



Thèse présentée à

l'Université Clermont Auvergne, France

École Doctorale des Sciences Pour l'Ingénieur

Unité de recherche: Laboratoire d'Informatique, de Modélisation et  
d'Optimisation des Systèmes (LIMOS)

---

---

# Structural and algorithmic aspects of identification problems in graphs

---

---

Auteur : Dipayan CHAKRABORTY

Directrice de thèse : Annegret WAGLER

Co-encadrant : Florent FOUCAUD

Co-encadrant : Michael HENNING

Date de soutenance : 09 decembre 2024

pour obtenir le grade de

Docteur (spécialité : informatique)

Devant le jury composé de :

**Président du jury:** Paul DORBEC, Professeur, Université de Caen-Normandie

Annegret WAGLER	Professeure, Université Clermont Auvergne	Directrice
Florent FOUCAUD	Maître de conférences, Université Clermont Auvergne	Co-encadrant
Michael HENNING	Professeur, University of Johannesburg	Co-encadrant
Paul DORBEC	Professeur, Université de Caen-Normandie	Examineur
Ralf KLASING	Directeur de Recherche, CNRS, Université de Bordeaux	Examineur
Tero LAIHONEN	Professeur, University of Turku	Rapporteur
Arnaud PÊCHER	Professeur, Université de Bordeaux	Rapporteur



Thèse présentée à

l'Université Clermont Auvergne, France

École Doctorale des Sciences Pour l'Ingénieur

Unité de recherche: Laboratoire d'Informatique, de Modélisation et  
d'Optimisation des Systèmes (LIMOS)

---

---

# Algorithmique et structure des problèmes d'identification

---

---

Auteur : Dipayan CHAKRABORTY

Directrice de thèse : Annegret WAGLER

Co-encadrant : Florent FOUCAUD

Co-encadrant : Michael HENNING

Date de soutenance : 09 decembre 2024

pour obtenir le grade de

Docteur (spécialité : informatique)

Devant le jury composé de :

**Président du jury:** Paul DORBEC, Professeur, Université de Caen-Normandie

Annegret WAGLER	Professeure, Université Clermont Auvergne	Directrice
Florent FOUCAUD	Maître de conférences, Université Clermont Auvergne	Co-encadrant
Michael HENNING	Professeur, University of Johannesburg	Co-encadrant
Paul DORBEC	Professeur, Université de Caen-Normandie	Examineur
Ralf KLASING	Directeur de Recherche, CNRS, Université de Bordeaux	Examineur
Tero LAIHONEN	Professeur, University of Turku	Rapporteur
Arnaud PÊCHER	Professeur, Université de Bordeaux	Rapporteur

# Acknowledgement

I would like to take this opportunity to express my sincere gratitude to all those who have contributed to the successful completion of this thesis. This journey has been both challenging and rewarding, and I could not have accomplished it without the unwavering support and guidance of many individuals and institutions.

First and foremost, I wish to extend my deepest appreciation to my thesis advisors, Prof. Annegret Wagler, Prof. Florent Foucaud and Prof. Michael Henning, whose invaluable mentorship, patience, and insightful feedback have played a crucial role in shaping my research. Their expertise and constant encouragement have guided me through every stage of this work, helping me refine my ideas and overcome challenges along the way. I am also immensely grateful to the other members of the jury, Prof. Paul Dorbec, Prof. Ralf Klasing, Prof. Tero Laihonen and Prof. Arnaud Pêcher, for their thoughtful ideas and constructive suggestions. Their insightful feedback has greatly helped to improve the final version of my thesis and to achieve higher standards of academic excellence.

My heartfelt appreciation goes to my parents, my wife, other members of my family, my friends and well-wishers for their unconditional love, support and patience. Their encouragement has been the greatest source of my motivation. A special mention of thankfulness and gratitude to Sagnik Sen, a longtime friend and colleague, and Soumen Nandi, another friend and colleague whom I have also known for several years now. The last-two gentlemen had been instrumental in helping me restart my career in academic research, a profession I always loved and wanted to be part of but somehow got derailed from along the way. Their support and willingness to work with me on (graph theoretic) problems that I did not have much experience in till then, and in many ways, teaching me the very basics of the subject, had been the first steps to me accomplishing my PhD today. I am equally thankful to Sandeep R.B. for hiring me as a contractual researcher at IIT Dharwad which basically marked the official start of my research journey in graph theory. Needless to say, my PhD is a product of this very important phase of my life during which the support of these three individuals had been enormous and selfless.

Apart from the above-mentioned people, I would also like to express my gratitude to all my other research collaborators with whom I have had the pleasure to work on several research topics during my PhD and whose ideas, hard work and insights have directly contributed to the success of this thesis. This journey has been even more memorable not only because of the research but also because of the meaningful connections I formed with colleagues, collaborators and fellow PhDs like Tapas, Pavan, Zin, Supraja, Taruni, Clara, Armel, Chijia, Idris, Imran, Romain, Javad, Antoine, Tuomo, Anni, Jan, Nikola, Diptapriyo, Pratik and many others. Through several enjoyable activities, excursions, tours, games, and other shared experiences, I built friendships and bonds that made this experience truly special.

Furthermore, I am deeply thankful to Université Clermont Auvergne for providing me with the opportunity to carry out my PhD and for the necessary facilities to conduct research with an utmost amount of freedom. I sincerely acknowledge the several funding opportunities and travelling assistances that I have received from my university in order to attend conferences, seminars, workshops, research visits, summer schools and several other scientific events which have only enriched my PhD experience by expanding my scientific network, connections and knowledge. In this connection, I am also thankful to Prof. Tero Laihonen and his research team including Tuomo and Anni at the University of Turku for hosting my very fruitful research visit to Finland during the summer of 2023. Also, special thanks to the members of my laboratory, LIMOS, for their company, assistance, friendship and the beautiful memories that I have built with them during my PhD days.

Last but not the least, I would like to acknowledge the French National Research Agency for financing of my PhD by means of a public grant that is part of the “Investissements d’Avenir” of the IMobS3 Laboratory of Excellence (ANR-10-LABX-0016); and by the French government IDEX-ISITE initiative 16-IDEX-0001 (CAP 20-25) and the International Research Center “Innovation Transportation and Production Systems” of the I-SITE CAP 20-25.

# Résumé

Les problèmes d'identification dans les graphes concernent en général la *séparation* de sommets. Une des formulations les plus étudiées utilise des *ensembles dominants*. Un *ensemble dominant* est un sous-ensemble de sommets  $S$  d'un graphe  $G$  tel que tout sommet non inclus dans  $S$  doit avoir un voisin dans  $S$ . Ces ensembles permettent de *séparer* des paires de sommets en trouvant un ensemble dominant  $S$  où toute paire de sommets distincts a des ensembles de voisins distincts dans  $S$ .

Dans cette thèse, nous étudions des problèmes d'identification dans les graphes basés sur les ensembles dominants et *totalelement dominants* (une version plus forte). En variant les types de voisinages distincts qui séparent les sommets, nous analysons quatre propriétés de séparation (dont l'une est nouvellement introduite). Combinées avec les deux propriétés de domination, nous étudions huit ensembles de séparation-domination (souvent appelés *codes*). Nous examinons les  $X$ -codes, où  $X$  appartient à  $\{LD, LTD, ID, ITD, OD, OTD, FD, FTD\}$ , abréviations des noms des codes. L'objectif est de minimiser la cardinalité des codes d'un graphe donné, et nous appelons cette cardinalité minimale le *nombre de code* (ou  *$X$ -nombre*).

La thèse présente des résultats structurels et algorithmiques. Dans la partie structurelle, nous étudions les bornes générales sur les huit codes et comparons par paires leurs nombres de code afin de fournir un schéma complet de leurs relations. Nous étudions également des conjectures de la littérature sur les bornes des nombres de code. Pour un graphe  $G$ , les bornes supérieures sont de la forme  $f(G) \cdot n$ , où  $n$  est le nombre de sommets, et  $f(G)$  est une fonction du graphe.

Sur les graphes sans jumeaux, nous examinons une conjecture de borne supérieure avec  $f(G) = 1/2$  pour  $X = LD$ , et  $f(G) = 2/3$  pour  $X = LTD$ . Pour  $X = LD$ , nous prouvons que la conjecture est vraie pour les graphes sous-cubiques et les graphes en blocs. Pour  $X = LTD$ , la conjecture est vérifiée pour les graphes split, cobipartis, en blocs et sous-cubiques. Pour  $X = ID$ , nous prouvons une borne supérieure sur les graphes sans triangles avec  $f(G) = (d - 1)/d$ , où  $d$  est le degré maximum de  $G$ . Pour  $X = OTD$ , nous trouvons un résultat exact pour les cycles avec  $f(G)$  environ égal à  $2/3$  ; pour les graphes sans cycles de longueur 4, nous prouvons une borne supérieure avec  $f(G) = (2d - 1)/2d$ . Pour les graphes en blocs, nous établissons des bornes supérieures et inférieures des  $X$ -nombres pour  $X = LD, ID$  et  $OTD$ . Pour  $X = OD$ , nous trouvons des valeurs exactes pour des classes de graphes comme les semi-graphes (half-graphs) et les graphes araignées sans tête. Enfin, pour  $X = FD$  et  $FTD$ , nous déterminons les valeurs exactes des nombres de code pour les chemins, cycles, semi-graphes et graphes araignées sans tête.

Dans la partie algorithmique, nous prouvons que pour les nouveaux  $X$ -codes introduits, c'est-à-dire pour  $X = OD, FD$  et  $FTD$ , trouver un  $X$ -code minimal est un problème **NP**-complet. Nous prouvons également qu'il est **NP**-difficile de décider si la différence entre les  $OD$ - et  $OTD$ -nombres (ou les  $FD$ - et  $FTD$ -nombres) est d'au plus 1 pour un graphe donné. Enfin, pour le problème de trouver un code  $LD$  minimal, déjà prouvé **NP**-complet, nous examinons des paramètres naturels et structurels. Nous fournissons une borne inférieure sur les temps d'exécution en fonction de la taille de la solution et prouvons un résultat d'incompressibilité lié à l'ordre du graphe. Pour les paramètres nombre de couverture par sommets, nombre de couverture par jumeaux et distance à une clique, nous proposons des algorithmes sur-exponentiels. Pour le paramètre largeur arborescente, nous fournissons un algorithme doublement exponentiel, optimal sous l'ETH. Pour la diversité de voisinage et le nombre cyclomatique comme paramètres, nous montrons que le problème possède un noyau linéaire.

# Abstract

Identification problems in graphs are those which, in general, deal with *separating* one vertex from another. To that end, one of the most well-studied formulations of such problems is by using *dominating sets* in graphs. A *dominating set* is a vertex subset  $S$  of a graph  $G$  such that any vertex not in  $S$  must have a neighbor in  $S$ . Using such sets as a combinatorial tool, pairs of vertices can be *separated* by finding a dominating set such that any two distinct vertices have distinct sets of neighbors in the set.

In this thesis, we study such identification problems in graphs based on dominating and *total-dominating sets* (a stronger variant of dominating sets). Also varying the types of (distinct) neighborhoods by which the vertices are separated, we look at four separating properties, one of which is newly introduced. In combination with the two dominating properties therefore, we study at eight separating-dominating sets (often called *codes*) in graphs. In short, we study  $X$ -codes of graphs, where  $X$  belongs to the set  $\{LD, LTD, ID, ITD, OD, OTD, FD, FTD\}$  of abbreviations of the original code names. The objective of such studies is to minimize the cardinalities of the codes and we call such a minimum cardinality to be a *code number* (or the  *$X$ -number*).

The thesis has both structural and algorithmic results. In the structural part, we study general bounds on all the eight codes and compare pairwise their code numbers to provide a complete scheme of their relations. We also study several conjectures from the literature on bounds of code numbers. For a graph  $G$ , the upper bounds we study are of the form  $f(G) \cdot n$ , where  $n$  is the number of vertices of a graph and  $f(G)$  is some number that is a function of the graph  $G$ .

On twin-free graphs, we study such an upper bound conjecture with  $f(G) = 1/2$  for  $X = LD$  and with  $f(G) = 2/3$  for  $X = LTD$ . For  $X = LD$ , we prove the conjecture to be true and tight for subcubic and block graphs. Moreover, for  $X = LTD$ , we prove the corresponding conjecture to be true and tight for split, cobipartite, block and subcubic graphs. For  $X = ID$ , we prove a tight upper bound conjecture on triangle-free graphs with  $f(G) = (d - 1)/d$ , where  $d$  is the maximum degree of a graph  $G$ . For  $X = OTD$ , we prove an exact result for cycles with  $f(G)$  being about  $2/3$ ; and for four-cycle-free graphs, we prove a tight upper bound with  $f(G) = (2d - 1)/2d$ , where  $d$  is again the maximum degree of  $G$ . Moreover, for block graphs, we establish both upper bounds (including a conjecture) and lower bounds on  $X$ -numbers for  $X = LD$ ,  $ID$  and  $OTD$  in terms of the number of vertices and blocks (maximal 2-connected subgraphs). For  $X = OD$ , we study and find exact values of the  $OD$ -numbers for several graph classes like half-graphs, headless spiders and suns. Finally, for  $X = FD$  and  $FTD$ , we again find the exact values of the respective code numbers for paths, cycles, half-graphs and headless spiders.

In the algorithmic part of the thesis, we prove that for the newly introduced  $X$ -codes, that is for  $X = OD$ ,  $FD$  and  $FTD$ , finding a minimum  $X$ -code is **NP**-complete. We also prove that, despite the  $OD$ - and the  $OTD$ -numbers (respectively, the  $FD$ - and the  $FTD$ -numbers) differing with each other by at most 1, it is in general **NP**-hard to decide if such a difference exists on a given graph. Finally, with the problem of finding a minimum  $LD$ -code being already known to be **NP**-complete, we look at the problem parameterized by some natural and structural parameters. We provide a tight lower bound on algorithmic running times in terms of the solution size and also prove an incompressibility result in terms of the order of the graph. With vertex cover number, twin-cover number and distance to clique, we provide algorithms running in slightly super-exponential times. Moreover, when parameterized by treewidth, we provide a double-exponential algorithm that is also tight under the **ETH**. With neighborhood diversity and feedback edge set number as parameters, we show that the problem has a linear kernel.

# Contents

<b>1</b>	<b>Introduction</b>	<b>13</b>
1.1	Monitoring systems and related codes . . . . .	13
1.1.1	Locating-dominating codes . . . . .	16
1.1.2	Identifying codes . . . . .	18
1.1.3	Open-separating total-dominating codes . . . . .	19
1.1.4	Full-separating total-dominating codes . . . . .	20
1.1.5	More codes based on domination . . . . .	21
1.2	Other applications of codes . . . . .	23
1.3	Organization of the thesis . . . . .	23
1.4	Other work carried out during PhD . . . . .	24
<b>2</b>	<b>Preliminaries and overview</b>	<b>25</b>
2.1	Notations and terminologies . . . . .	25
2.1.1	General mathematics . . . . .	25
2.1.2	Graph theory . . . . .	26
2.1.3	Graph families . . . . .	31
2.1.4	Computational complexity . . . . .	37
2.1.5	Parameterized complexity . . . . .	40
2.2	Separating sets revisited . . . . .	44
2.2.1	Twins and separators . . . . .	45
2.2.2	Separation of vertices . . . . .	46
2.3	A unified system of notations for codes . . . . .	50
2.4	Other prevalent terminologies related to codes . . . . .	51
2.5	Literature survey . . . . .	51
2.5.1	Bounds on code numbers in specific graph families . . . . .	51
2.5.2	Computational complexity of X-CODE . . . . .	58
<b>3</b>	<b>A unified study of codes</b>	<b>63</b>
3.1	Hypergraph representation of codes . . . . .	63
3.2	Polyhedra associated with codes . . . . .	65
3.3	Comparisons among codes . . . . .	66
3.4	General bounds on codes . . . . .	73
3.4.1	General upper bounds . . . . .	74
3.4.2	General lower bounds . . . . .	75
3.5	Conclusion . . . . .	84
<b>I</b>	<b>Structural aspects: Bounds on code numbers</b>	<b>87</b>
<b>4</b>	<b>Location in graphs</b>	<b>89</b>
4.1	Location with domination . . . . .	90
4.1.1	Preliminary results . . . . .	91
4.1.2	Block graphs . . . . .	91

4.1.3	Subcubic graphs . . . . .	101
4.2	Location with total domination . . . . .	121
4.2.1	Cobipartite graphs . . . . .	121
4.2.2	Split graphs . . . . .	123
4.2.3	Block graphs . . . . .	124
4.2.4	Subcubic graphs . . . . .	131
4.3	Conclusion . . . . .	135
4.3.1	LD-codes of block graphs . . . . .	135
4.3.2	LD-codes of subcubic graphs . . . . .	136
4.3.3	Two-thirds conjecture for LTD-codes . . . . .	136
<b>5</b>	<b>Closed separation in graphs</b>	<b>139</b>
5.1	Block graphs . . . . .	140
5.1.1	Upper bounds on ID-numbers of block graphs . . . . .	141
5.1.2	Lower bounds on ID-numbers of block graphs . . . . .	142
5.2	Graphs of a given maximum degree . . . . .	146
5.2.1	Paths and cycles . . . . .	147
5.2.2	Trees of given maximum degree . . . . .	148
5.2.3	Triangle-free graphs of given maximum degree . . . . .	166
5.2.4	Beyond triangle-free graphs . . . . .	189
5.3	Conclusion . . . . .	190
5.3.1	ID-codes of block graphs . . . . .	190
5.3.2	ID-codes of trees of given maximum degree . . . . .	190
5.3.3	ID-codes of triangle-free graphs of given maximum degree . . . . .	191
<b>6</b>	<b>Open separation in graphs</b>	<b>193</b>
6.1	Open separation with total domination . . . . .	193
6.1.1	Block graphs . . . . .	194
6.1.2	$n$ -cycles . . . . .	201
6.1.3	$C_4$ -free graphs with bounded maximum degree . . . . .	203
6.2	Open separation with domination . . . . .	217
6.2.1	Preliminary results . . . . .	217
6.2.2	OD-numbers of some graph families . . . . .	218
6.3	Conclusion . . . . .	224
6.3.1	OTD-codes of block graphs . . . . .	224
6.3.2	OTD-codes of $C_4$ -free graphs of given maximum degree . . . . .	225
6.3.3	OD-codes of graphs . . . . .	225
<b>7</b>	<b>Full separation in graphs</b>	<b>227</b>
7.1	Preliminary results . . . . .	227
7.2	FD- and FTD-numbers of some graph families . . . . .	229
7.2.1	Paths and cycles . . . . .	229
7.2.2	Half-graphs . . . . .	231
7.2.3	Headless spiders . . . . .	233
7.3	Conclusion . . . . .	235
<b>II</b>	<b>Algorithmic aspects: Complexities and parameterized algorithms</b>	<b>237</b>
<b>8</b>	<b>NP-hardness results</b>	<b>239</b>
8.1	Open separation . . . . .	241
8.2	Full separation . . . . .	246
8.3	Conclusion . . . . .	250

<b>9</b>	<b>Location domination as a parameterized problem</b>	<b>251</b>
9.1	LD-CODE parameterized by Solution Size . . . . .	253
9.2	Incompressibility of LD-CODE . . . . .	257
9.3	LD-CODE parameterized by treewidth . . . . .	259
9.3.1	Upper bound . . . . .	260
9.3.2	Lower bound . . . . .	265
9.4	LD-CODE parameterized by vertex cover number . . . . .	269
9.5	Other structural parameters . . . . .	275
9.5.1	Extending Theorem 9.5 for twin-cover number and distance to clique . . . . .	275
9.5.2	A linear kernel for LD-CODE parameterized by neighborhood diversity . . . . .	276
9.6	Linear kernel for LD-CODE parameterized by feedback edge set number . . . . .	276
9.7	Conclusion . . . . .	281
<b>10</b>	<b>Conclusion and perspectives</b>	<b>283</b>
10.1	Summary of our work . . . . .	283
10.1.1	Part I. Structural aspects . . . . .	283
10.1.2	Part II. Algorithmic aspects . . . . .	285
10.2	Perspectives and future research . . . . .	286
10.2.1	Related identification problems on graphs . . . . .	287
10.2.2	Further classification of codes and potential future problems . . . . .	288
10.3	Concluding remarks . . . . .	290



# Résumé étendu

Dans cette thèse, nous étudions les problèmes d'identification dans les graphes. Ces problèmes consistent à *séparer* les sommets d'un graphe entre eux. L'une des formulations les plus étudiées de ces problèmes d'identification repose sur l'utilisation des *ensembles dominants* dans les graphes. Un *ensemble dominant* est un sous-ensemble de sommets  $S$  d'un graphe  $G$  tel que tout sommet qui n'appartient pas à  $S$  possède au moins un voisin dans  $S$ . Dans ces problèmes d'identification basés sur la domination, l'objectif est de trouver un ensemble dominant  $S$  approprié dans un graphe  $G$  de sorte que deux sommets distincts de  $G$  aient des ensembles de voisins distincts dans  $S$ . Un tel ensemble dominant est également appelé un *ensemble séparateur* et est dit être doté de la propriété de *séparer* les sommets d'un graphe.

Dans ce travail, nous étudions principalement quatre types différents d'ensembles séparateurs, à savoir, ceux ayant les propriétés de *localisation* (*location*), *séparation fermée* (*closed-separation*), *séparation ouverte* (*open-separation*) et *séparation complète* (*full-separation*). Combinés avec des ensembles dominants et totalement dominants dans les graphes, cela donne lieu à huit types différents de codes, à savoir :

- (1) les *codes de domination localisants* (*locating-dominating codes*, ou *LD-codes* en abrégé)
- (2) les *codes de domination totale localisants* (*locating-total-dominating codes*, ou *LTD-codes* en abrégé),
- (3) les *codes identifiants* (*identifying codes*, ou *ID-codes* en abrégé),
- (4) les *codes de domination totale identifiants* (*identifying total-dominating codes*, ou *ITD-codes* en abrégé),
- (5) les *codes de domination à séparation ouverte* (*open-separating dominating codes*, ou *OD-codes* en abrégé),
- (6) les *codes de domination totale à séparation ouverte* (*open-separating total-dominating codes*, ou *OTD-codes* en abrégé),
- (7) les *codes de domination à séparation complète* (*full-separating dominating codes*, ou *FD-codes* en abrégé) et
- (8) les *codes de domination totale à séparation complète* (*full-separating total-dominating codes*, ou *FTD-codes* en abrégé).

En bref, nous appelons les codes mentionnés ci-dessus les  $X$ -codes pour  $X \in \text{CODES} = \{\text{LD}, \text{LTD}, \text{ID}, \text{ITD}, \text{OD}, \text{OTD}, \text{FD}, \text{FTD}\}$ . Trois de ces codes, à savoir les OD-, FD- et FTD-codes ont été introduits assez récemment dans la littérature des problèmes d'identification. Étant donné tout  $X \in \text{CODES}$ , tant d'un point de vue combinatoire qu'algorithmique, nous nous intéressons principalement à la recherche de la cardinalité minimale d'un  $X$ -code d'un graphe  $G$ . Une telle cardinalité minimale est appelée le  $X$ -nombre de  $G$  et est notée  $\gamma^X(G)$ . L'étude menée dans cette thèse commence par une comparaison des huit  $X$ -nombres entre eux. Pour réaliser cette comparaison, nous utilisons une reformulation canonique du problème de détermination des  $X$ -nombres d'un graphe en un problème équivalent de recherche du nombre de couverture d'un hypergraphe correspondant (appelé  $X$ -hypergraphe). Par la suite, les hyperarêtes de ces  $X$ -hypergraphes fournissent un cadre naturel pour les comparaisons entre ces nombres de codes.

Nous utilisons également un autre outil, à savoir une reformulation généralisée du théorème de Bondy [Bondy, J. A.: Induced Subsets. Journal of Combinatorial Theory **12**(B), pp. 201-202 (1972)]. Grâce à cet outil, nous montrons qu'il est également possible d'inverser les ordres de ces comparaisons (quoique avec un facteur multiplicatif). Cela nous permet d'obtenir une vision globale des huit codes étudiés dans cette thèse et de la manière dont leurs nombres respectifs se comparent entre eux. Nous fournissons également des bornes générales supérieures et inférieures pour ces codes. De plus, pour chaque  $X \in \text{CODES}$ , nous donnons une description générique de tous les graphes dont

le X-nombre atteint cette borne inférieure générale (qui est déjà connue dans la littérature comme étant logarithmique en fonction de l'ordre du graphe).

En ce qui concerne l'étude liée à chaque code, la structure de cette thèse peut être globalement divisée en deux parties. Dans la Partie I, nous présentons les résultats structurels concernant les bornes supérieures et inférieures sur les nombres de codes des graphes, avec un accent particulier sur l'étude de ces codes dans des classes de graphes bien connues. Dans la Partie II, nous étudions les aspects algorithmiques relatifs à X-CODE qui est la version décisionnelle du problème de minimisation d'un X-code. En d'autres termes, étant donné un graphe  $G$  et un paramètre  $k$ , le problème X-CODE demande s'il est possible de trouver un X-code de  $G$  d'ordre au plus  $k$ . Dans notre étude algorithmique des X-codes, nous nous concentrons principalement sur les codes pour  $X \in \{\text{LD}, \text{OD}, \text{FD}, \text{FTD}\}$ . Pour les trois derniers codes, nous nous concentrons sur la détermination de la complexité du problème, et pour  $X = \text{LD}$  (déjà connu comme étant NP-complet dans la littérature), nous explorons certains aspects de la complexité paramétrée de LD-CODE.

**Partie I. Aspects structurels : Bornes sur les nombres de codes.** En ce qui concerne les aspects structurels des problèmes, nous étudions chaque type d'ensemble séparant dans un chapitre distinct et analysons les nombres de codes correspondants, à la fois de manière générale et sur certaines classes spécifiques de graphes.

Dans le chapitre 4, nous examinons la propriété de séparation par localisation et l'étudions en combinaison avec la domination (c'est-à-dire les LD-codes) et la domination totale (c'est-à-dire les LTD-codes). Notre étude des LD-codes dans ce chapitre se concentre sur les classes des graphes en blocs et des graphes sous-cubiques. Pour les graphes en blocs, nous établissons des bornes supérieures et inférieures générales et serrées sur leurs nombres de LD-codes. De plus, nous vérifions, pour les graphes en blocs, la conjecture suivante, d'abord postulée dans [Garijo, D., González, A., Márquez A.: The difference between the metric dimension and the determining number of a graph. *Applied Mathematics and Computation* **249**(1), pp. 487–501 (2014)], puis reformulée sous une forme légèrement plus forte dans [Foucaud, F., Henning, M.A.: Location-domination and matching in cubic graphs. *Discrete Mathematics* **339**(4), pp. 1221–1231 (2016)]. Nous prouvons que ladite conjecture est à la fois vraie et atteignable pour les graphes en blocs.

**Conjecture.** *Soit  $G$  un graphe sans jumeaux et sans sommets isolés de  $n$  sommets. Alors, nous avons :*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

Sur les graphes sous-cubiques, nous poursuivons notre étude de la borne supérieure  $n$ -moitié, c'est-à-dire la conjecture mentionnée ci-dessus, et nous démontrons qu'elle est également valide pour cette classe de graphes. De plus, nous élargissons la portée de cette conjecture pour les graphes sous-cubiques en prouvant qu'elle reste vraie même lorsque ces graphes possèdent des jumeaux fermés et des jumeaux ouverts de degré 3, à l'exception des graphes  $K_4$  et  $K_{3,3}$ . Ce résultat apporte également une réponse positive aux questions soulevées dans [Foucaud, F., Henning, M.A.: Location-domination and matching in cubic graphs. *Discrete Mathematics* **339**(4), pp. 1221–1231 (2016)], concernant la véracité de cette conjecture pour les graphes cubiques avec la présence de jumeaux. Cependant, nous montrons que la conjecture ne s'étend pas à tous les graphes réguliers avec jumeaux en général, puisque la borne échoue pour les graphes  $r$ -réguliers avec  $r \geq 4$ .

Dans la deuxième partie du chapitre 4, nous étudions les codes de domination totale localisante et établissons des bornes sur les LTD-nombres de plusieurs classes de graphes, notamment les graphes cobipartis, les graphes scindés, les graphes en blocs et les graphes sous-cubiques. Plus précisément, nous examinons la conjecture suivante apparaissant dans [Foucaud, F., Henning, M.A.: Locating-total dominating sets in twin-free graphs: a conjecture. *The Electronic Journal of Combinatorics* **23**(3), P3.9 (2016)] pour toutes ces classes de graphes et prouvons qu'elle est vraie et atteignable.

**Conjecture.** *Tout graphe  $G$  sans jumeaux et sans sommets isolés d'ordre  $n$  satisfait*

$$\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n.$$

Nos résultats pour certaines classes de graphes sont en réalité encore plus forts que ceux énoncés dans ladite conjecture. Par exemple, au lieu de la borne supérieure conjecturée des deux tiers de l'ordre du graphe  $\binom{2n}{3}$ , nous montrons que cette borne baisse à la moitié de  $n$  pour les graphes cobipartis et qu'elle est strictement inférieure à  $\frac{2n}{3}$  pour les graphes scindés. De plus, nous démontrons que cette borne supérieure (initialement formulée pour les graphes sans jumeaux) reste valide pour les graphes sous-cubiques, même lorsque ces derniers contiennent des jumeaux.

Dans le chapitre 5, nous étudions les codes identifiants sur différentes classes de graphes. Ici aussi, nous établissons des bornes supérieures et inférieures précises sur les ID-nombres des graphes en blocs. En particulier, nous examinons la conjecture suivante sur la borne supérieure des ID-nombres des graphes en blocs introduite dans [Argiroffo, G.R., Bianchi, S.M., Lucarini Y., Wagler A.K.: On the identifying code number of block graphs. In: Proceedings of ICGT 2018, Lyon, France (2018)], laquelle est exprimée en fonction du nombre de blocs du graphe.

**Conjecture.** *Le ID-nombre d'un graphe en blocs sans jumeaux fermés est majoré par son nombre de blocs.*

Ensuite, dans le même chapitre, nous poursuivons notre étude des ID-nombres des graphes en fonction de leur degré maximum et de leur ordre. Cette approche est motivée par la conjecture suivante, apparue pour la première fois dans [Foucaud, F., Klasing, R., Kosowski, A., Raspaud, A.: On the size of identifying codes in triangle-free graphs. Discrete Applied Mathematics **160**, pp. 1532-1546 (2012)], que nous démontrons pour les arbres et les graphes sans triangles.

**Conjecture.** *Il existe une constante  $c$  telle que pour tout graphe connexe sans jumeaux fermés, d'ordre  $n \geq 2$  et de degré maximum  $\Delta \geq 2$ , nous avons :*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n + c.$$

Tout d'abord, nous étudions la conjecture ci-dessus pour les arbres. Comme la conjecture est déjà prouvée pour les chemins (avec  $\Delta = 2$ ) dans la littérature, nous nous concentrons sur le cas où  $\Delta \geq 3$ . Nous montrons que, à l'exception d'une petite famille  $\mathcal{T}_\Delta$  de graphes exceptionnels, la conjecture est vérifiée avec une constante  $c = 0$  pour les arbres avec  $\Delta \geq 3$ . De plus, nous caractérisons l'ensemble  $\mathcal{T}_\Delta$  des arbres exceptionnels pour lesquels  $c > 0$ , et prouvons que pour  $\Delta = 3$ , l'ensemble  $\mathcal{T}_\Delta$  est composé de 12 arbres de degré maximum 3 et de diamètre au plus 6; et pour  $\Delta \geq 4$ , nous avons  $\mathcal{T}_\Delta = \{K_{1,\Delta}\}$ . Enfin, nous démontrons également une relation intéressante entre le ID-nombre et le nombre de domination d'un arbre, et nous prouvons que leur somme ne peut jamais dépasser l'ordre de l'arbre.

Nous poursuivons ensuite l'étude de la dernière conjecture pour les graphes sans triangle et prouvons à nouveau sa validité dans ce cas. Ici aussi, nous nous concentrons sur le cas où  $\Delta \geq 2$  (puisque le résultat pour  $\Delta = 2$  est déjà connu dans la littérature). Notre résultat sur ladite conjecture pour les arbres sert de point de départ pour la preuve de cette même conjecture dans le cas des graphes sans triangle. De plus, au cours de cette étude, nous démontrons une version généralisée du théorème de Bondy sur les sous-ensembles induits, que nous utilisons comme outil dans nos démonstrations. Nous exploitons également notre résultat principal sur les graphes sans triangle pour établir la borne supérieure  $\left( \frac{\Delta-1}{\Delta} \right) n + 1/\Delta + 4t$  pour les graphes pouvant être rendus sans triangle par la suppression de  $t$  arêtes.

Dans le chapitre 6, nous étudions la propriété de séparation ouverte combinée à la domination (c'est-à-dire les OD-codes) et à la domination totale (c'est-à-dire les OTD-codes). Dans la première partie du chapitre, nous examinons les OTD-codes dans certaines classes de graphes. Tout d'abord, nous considérons une fois de plus les graphes en blocs et obtenons des bornes supérieures et inférieures précises pour les OTD-nombres de ces graphes. Notre borne supérieure généralise un résultat de [Foucaud, F., Ghareghani, N., Roshany-Tabrizi, A., Sharifani, P.: Characterizing extremal graphs for open neighbourhood location-domination. Discrete Applied Mathematics **302**, pp. 76-79 (2021)] sur les semi-graphes (notons que les semi-graphes  $P_2$  et  $P_4$  sont également des graphes en blocs).

Dans ce chapitre, nous étudions également les OTD-codes des cycles et démontrons un résultat exact pour ces nombres de codes en fonction de l'ordre du cycle. Ensuite, nous analysons les bornes des OTD-nombres des graphes en fonction de leur degré maximum et de leur ordre. Plus précisément, nous démontrons la borne supérieure suivante pour les graphes sans  $C_4$ .

**Théorème.** *Pour  $\Delta \geq 3$  un entier fixé, si  $G$  est un graphe connexe sans jumeaux ouverts d'ordre  $n \geq 5$  ne contenant aucun 4-cycle et satisfaisant  $\Delta(G) \leq \Delta$ , alors*

$$\gamma^{\text{OTD}}(G) \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n,$$

*sauf dans un cas exceptionnel où  $G$  est isomorphe à une  $\Delta$ -étoile subdivisée, auquel cas*

$$\gamma^{\text{OTD}}(G) = \left( \frac{2\Delta}{2\Delta + 1} \right) n.$$

Nous prouvons d'abord le théorème ci-dessus pour les arbres (qui sont aussi sans 4-cycle), puis nous utilisons ce résultat pour démontrer le théorème pour tous les graphes sans 4-cycle.

Dans la deuxième partie du même chapitre, nous nous intéressons aux OD-codes des graphes, qui ont été introduits assez récemment dans [Chakraborty, D., Wagler, A.K.: Open-Separating Dominating Codes in Graphs. Combinatorial Optimization (ISCO 2024) Lecture Notes in Computer Science **14594**, pp. 137-151 (2024)]. Nous montrons que les OD-codes sont étroitement liés aux OTD-codes, dans la mesure où leurs nombres respectifs ne diffèrent au plus que de 1. Cela nous motive à étudier les OD-nombres (en comparaison avec les OTD-nombres déjà connus dans la littérature) pour plusieurs familles de graphes, notamment les graphes complets, les couplages et leurs unions disjointes, les semi-graphes et les étoiles subdivisées comme sous-familles des graphes bipartis, certaines sous-familles de graphes scindés (notamment certains soleils fins), et les araignées sans tête. Nos résultats sur les OD-nombres pour toutes ces familles de graphes sont exacts dans le sens où les bornes sont atteintes et donc les meilleures possibles. Nous étudions également les OD-codes sous un angle polyédral, en déterminant les OD-polyèdres des cliques, des couplages et des araignées sans tête épaisses et fines.

Dans le chapitre 7, nous étudions deux types de codes qui ont été récemment introduits dans [Chakraborty, D., Wagler, A.K.: On full-separating sets in graphs. Algorithms and Discrete Applied Mathematics (CALDAM 2025) Lecture Notes in Computer Science **15536**, pp. 73-84 (2025)] dans la littérature sur les problèmes d'identification. Plus précisément, nous analysons les FD-codes (séparation complète combinée avec la domination) et les FTD-codes (séparation complète combinée avec la domination totale) sur les graphes. Comme dans le cas des OD- et OTD-codes, nous constatons que les FD- et FTD-codes sont fortement liés, leurs nombres respectifs différant au plus de 1. Ainsi, nous étudions ces deux nombres de codes et les comparons sur plusieurs familles de graphes, telles que les chemins, les cycles, les semi-graphes et les araignées sans tête. Là encore, tous nos résultats concernant les FD- et FTD-nombres de codes sont exacts.

**Partie II. Aspects algorithmiques : Complexités et algorithmes paramétrés.** Les chapitres 8 et 9 sont consacrés à l'étude des aspects algorithmiques et de la complexité computationnelle des problèmes d'identification que nous abordons dans cette thèse.

Dans le chapitre 8, nous étudions la complexité algorithmique de OD-CODE, FD-CODE et FTD-CODE, qui sont les versions décisionnelles des problèmes consistant à déterminer respectivement le OD-nombre, le FD-nombre et le FTD-nombre des graphes. Nous prouvons que ces trois problèmes sont NP-complets. En dehors de ces problèmes, nous examinons également la question de savoir si les OD-nombres et les OTD-nombres (respectivement, les FD-nombres et les FTD-nombres) d'un graphe sont égaux ou non. Nous démontrons que, bien que ces nombres diffèrent d'au plus 1 dans chaque paire de types de codes considérée, il est en général NP-difficile de décider s'ils sont égaux ou non.

Dans le chapitre 9, nous nous intéressons aux aspects de complexité paramétrée de LD-CODE. Nous étudions le problème LD-CODE en le paramétrant à la fois par des paramètres naturels, comme

la taille de la solution et l'ordre du graphe, ainsi que par des paramètres structurels tels que la largeur arborescente, le nombre de couverture par sommets, le nombre de couverture par jumeaux, la distance à une clique, la diversité des voisinages et le nombre cyclomatique.

Nous étudions d'abord LD-CODE avec un graphe d'entrée ayant  $n$  sommets et un paramètre naturel correspondant à la taille de la solution, noté  $k$ . Nous démontrons que, sous l'hypothèse de temps exponentiel (ETH), le problème n'admet pas d'algorithme avec un temps d'exécution de  $2^{o(k^2)} \cdot n^{\mathcal{O}(1)}$ . Cela garantit essentiellement un temps d'exécution de l'ordre de  $2^{\mathcal{O}(k^2)} \cdot n^{\mathcal{O}(1)}$  pour ce problème (la borne supérieure de ce temps d'exécution provient du fait que LD-CODE possède un noyau de taille  $\mathcal{O}(2^k)$ , un résultat disponible à partir des travaux préliminaires de [Slater, P. J.: Dominating and reference sets in a graph. *Journal of Mathematical and Physical Sciences* **22**, pp. 445–455 (1988)]). Cette borne inférieure améliore deux bornes précédemment établies :  $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$  sous l'hypothèse  $W[2] \neq \text{FPT}$  dans [Barbero, F., Isenmann, L., Thiebaut, J.: On the distance identifying set meta-problem and applications to the complexity of identifying problems on graphs. *Algorithmica* **82**(8), pp. 2243–2266 (2020)] et  $2^{o(k \log k)}$  sous l'hypothèse ETH dans [Cappelle, M.R., Gomes, G.C.M., Santos, V.F.: Parameterized algorithms for locating-dominating sets. *arXiv:2011.14849 [cs.DS]* (2023)]. Nous examinons ensuite l'incompressibilité de LD-CODE et démontrons que, sauf si  $\text{NP} \subseteq \text{coNP}/\text{poly}$ , le problème n'admet pas de compression polynomiale de taille  $\mathcal{O}(n^{2-\epsilon})$  pour tout  $\epsilon > 0$ .

Nous explorons ensuite le problème LD-CODE paramétré par des paramètres structurels. Nous commençons par la largeur arborescente ( $\text{tw}$ ) et montrons que, sous l'hypothèse ETH, le problème n'admet pas d'algorithme à temps d'exécution  $2^{2^{o(\text{tw})}} \cdot \text{poly}(n)$ . Ce résultat place ce problème parmi la petite liste de problèmes de la classe  $\text{NP}$  qui nécessitent au moins un temps d'exécution doublement exponentiel en fonction de la largeur arborescente (les trois autres connus jusqu'à présent étant METRIC DIMENSION, STRONG METRIC DIMENSION et GEODETIC SET, étudiés dans [Foucaud, F., Galby, E., Khazaliya, L., Li, S., Mc Inerney, F., Sharma, R., Tale, T.: Tight (double) exponential bounds for NP-complete problems: Treewidth and vertex cover parameterizations. In: *Proceedings of ICALP 2024, Tallinn, Estonia* (2024)]). Nous prouvons également que cette borne inférieure est atteignable en proposant un algorithme de programmation dynamique résolvant le problème en un temps d'exécution de  $2^{2^{\mathcal{O}(\text{tw})}} \cdot n^{\mathcal{O}(1)}$ .

Ensuite, nous étudions LD-CODE paramétré par le nombre de couverture par sommets (vertex cover number, noté  $\text{vc}$ ) comme paramètre et concevons un algorithme de programmation dynamique légèrement sur-exponentiel pour ce problème, avec un temps d'exécution de  $2^{\mathcal{O}(\text{vc} \log \text{vc})} \cdot n^{\mathcal{O}(1)}$ . Nous montrons que ce résultat peut être étendu à d'autres paramètres tels que la distance à une clique et le nombre de couverture par jumeaux. Ensuite, nous démontrons que LD-CODE possède un noyau linéaire lorsqu'il est paramétré par la diversité des voisinages. Enfin, nous établissons également des noyaux linéaires existents pour ce problème lorsqu'il est paramétré par le nombre cyclomatique.

Nous concluons cette thèse par un chapitre final où nous résumons les résultats étudiés dans ce travail. Nous fournissons également des références à d'autres codes étudiés dans la littérature des problèmes d'identification mais non abordés dans cette thèse, afin d'attirer l'attention sur le fait que des techniques de preuve similaires à celles utilisées dans ce travail peuvent également être appliquées à plusieurs autres codes existants dans la littérature. À la fin de ce chapitre de conclusion, nous montrons aussi comment les définitions des huit codes étudiés dans cette thèse peuvent être généralisées encore plus loin pour les intégrer à une classification beaucoup plus large des problèmes d'identification basés sur la domination, dont plusieurs n'ont pas encore été considérés dans la littérature. Cela démontre les perspectives futures et les axes de recherche que l'on peut suivre concernant de tels problèmes.

# Chapter 1

## Introduction

### 1.1 Monitoring systems and related codes

Imagine that we have a task of putting a building with  $n$  rooms under surveillance such that each room of the building is monitored against the intrusion of an intruder. To build such a monitoring system, we are allowed to install some monitoring devices — like detectors or surveillance cameras — in some (but preferably not all) rooms of the building. In fact, as a cost-cutting measure, we would like to select the least number of rooms possible (again, preferably strictly less than  $n$ ) in which we would like to install these detectors. However, since we would like to monitor all rooms of the building with a number of detectors strictly less than the number of rooms, this implies that we need detectors that can monitor multiple rooms at a time — more than just the one they are installed in. Moreover, we would like our monitoring system to be *room-separating*, which is to say that the signal that the system generates to indicate the presence of an intruder must point to *exactly one* room in the building — where the intruder is supposed to be present. In a nutshell therefore, our problem of monitoring rooms in a building has the following requirements:

- (R1) The number of detectors must be the least possible.
- (R2) All rooms of the building must be monitored by detectors.
- (R3) The monitoring system must function in a room-separating manner thus producing unambiguous signals pointing to a unique room where the intruder would be present.

We next discuss how choosing the right kind of detectors that can monitor multiple rooms at a time and how distributing them out in a clever configuration across the rooms of the building can meet the above requirements of a monitoring system. Let us from now on refer to a room with a detector to be a *monitoring room* and one without a detector to be a *non-monitoring room*. See Figure 1.1 as a prototype example of a building for the discussions to follow in this chapter.

To express the above problem in a mathematical — more precisely, in a combinatorial — setting, we now digress a little to introduce some basic definitions of *graph theory*. We then transform the above real-life problem to be a problem on *graphs* and discuss the solutions to the latter problem. More technical and rigorous definitions on graphs shall follow in Chapter 2.

A *graph*  $G = (V(G), E(G))$  is a mathematical entity, where  $V(G)$  and  $E(G)$  are two sets. The set  $V(G)$  is called the *vertex set* of  $G$  and each element of  $V(G)$  is called a *vertex* of  $G$ . A vertex set can be thought of as a set of points in the three-dimensional space around us. The second set  $E(G)$  is called the *edge set* of  $G$  and each element of  $E(G)$  is called an *edge* of  $G$ . An edge of  $G$  can be thought of as a link or a connection between two distinct vertices of  $G$ . See Figure 1.2 for an example of a graph where the vertices are denoted by circles and edges are denoted by lines.

Given a graph  $G$ , if  $V(G) = \{v_1, v_2, \dots, v_n\}$ , where  $n$  is a positive integer, then any edge of  $G$  between two of its distinct vertices  $v_i$  and  $v_j$  is usually denoted by the symbol  $v_i v_j$ . In other words,

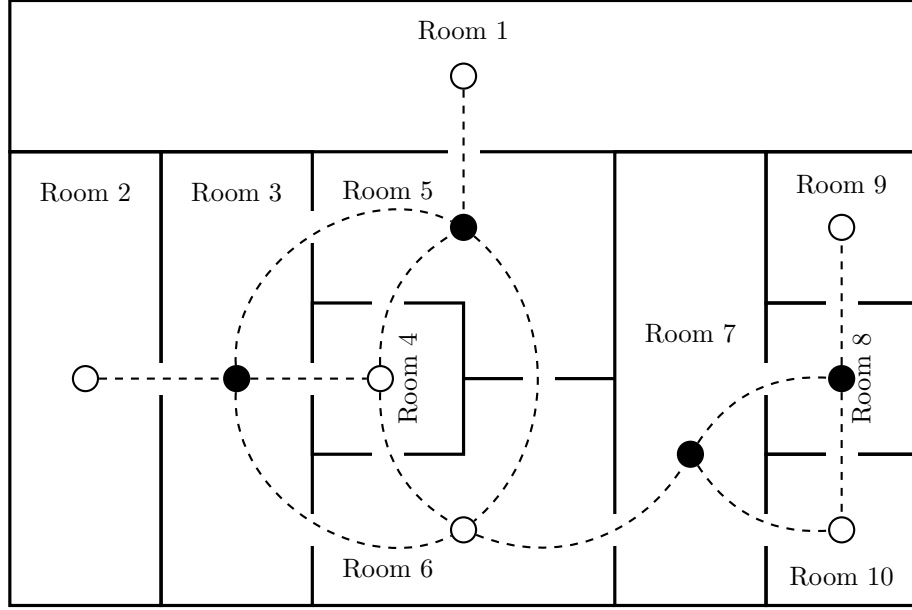


Figure 1.1: Schematic representation of a building plan. Each rectangle represents a room which can also be denoted as a vertex of a graph. An opening from one room to a neighboring room allows for signals from detectors to go through in order to sense the presence of an intruder in the adjacent room. This gives the notion of edges between vertices and indicates which neighboring rooms a detector can monitor. The black vertices represent the monitoring rooms (with detectors in them). As can be seen, each room has a detector either in itself or in a neighboring room with an entrance for signals to go through. Thus, all rooms of this building are monitored.

the edge set  $E(G)$  of  $G$  is some subset of the set  $\{v_i v_j : i \text{ and } j \text{ are integers and } 1 \leq i < j \leq n\}$ . Moreover, if  $v_i v_j$  is an edge of  $G$ , the vertices  $v_i$  and  $v_j$  are called *adjacent vertices* or *neighbors* in  $G$ . In addition, any edge  $v_i v_j$  of  $G$  is said to be *incident* with both  $v_i$  and  $v_j$ .

Given a graph  $G$ , any subset  $S$  of  $V(G)$  is called a *vertex subset* of  $G$  and any subset  $F$  of  $E(G)$  is called an *edge subset* of  $G$ . A graph  $H$  is called a *subgraph* of  $G$  if  $V(H) \subseteq V(G)$  and  $E(H) \subseteq E(G)$ . A subgraph (of a graph) is by itself a graph in its own right. Moreover, a subgraph  $H$  of  $G$  is called an *induced subgraph* of  $G$  if, for  $u, v \in V(H)$ , the pair  $uv$  is an edge of  $H$  if and only if  $uv$  is an edge of  $G$ . A graph  $G$  on  $n$  vertices such that  $V(G) = \{v_1, v_2, \dots, v_n\}$  and  $E(G) = \{v_i v_{i+1} : i \in [n-1]\}$  is called an  $n$ -*path*, or simply a *path* on  $n$  vertices. An  $n$ -path is usually denoted by  $P_n$ . Each of the vertices  $v_1$  and  $v_n$  is called an *endpoint* of the path  $P_n$  and the integer  $n-1$  is called the *length* of  $P_n$ . Given any two distinct vertices  $u, v$  of a graph  $G$ , let us consider each possible subgraph of  $G$  that, as a graph by itself, is a path with endpoints  $u$  and  $v$ . Then  $d_G(u, v)$  denotes the smallest length of all such paths considered. The quantity  $d_G(u, v)$  is called the *distance* between  $u$  and  $v$  in  $G$ .

With the above graph-theoretic definitions, we now return to our problem at hand. As shown in Figure 1.1, the graph-theoretic model of the above monitoring problem is realized by replacing the *building* by a *graph*  $G$  such that each *room* of the building is replaced by a corresponding *vertex* of  $G$  and, if any two rooms of the building are connected to each other — by way of having an entrance from one to the other — the corresponding vertices, say  $v_i$  and  $v_j$ , of these two rooms are made adjacent vertices in  $G$  by introducing the edge  $v_i v_j$  (Figure 1.2 shows the graph obtained by fully replacing the building by a corresponding graph). Moreover, the concept of a *monitoring room* in the building is replaced by labeling the vertex of  $G$  corresponding to the monitoring room to be a *code vertex* (the name “code” originates from the applications of such problems to coding theory — as pointed out in Section 1.2). In other words, the rooms corresponding to the code vertices

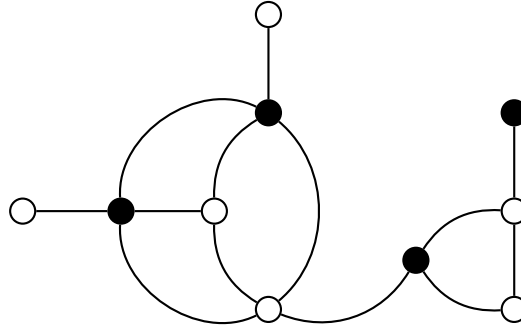


Figure 1.2: Graph  $G$  modelling the building plan in Figure 1.1. A diagrammatic representation of a graph is usually done by denoting the vertices by circles (or squares etc.) and denoting the edges by lines between certain pairs of vertices. In this example, the set of all black vertices is a code of  $G$  with each such black vertex being a code vertex of  $G$ .

are those equipped with a detector. Similarly, the concept of a *non-monitoring room* is replaced by labeling the corresponding vertex to be a *non-code vertex*. Let the set of all code vertices of a graph  $G$  be called a *code* of  $G$ . Even though the word “code” may not always intuitively imply so, according to our definition, a code here is simply a vertex subset of a graph (with each element of a code being called a “code vertex”).

Most surveillance devices prevalent today use either radiation or heat or other such detection techniques. In such techniques, a detector has a “monitoring sphere” of a certain radius  $r$  — or a topological ball of radius  $r$  centered around the detector in the three-dimensional Euclidean space — such that any intruder present within this sphere is so detected by the detector. Since we want each detector to monitor multiple rooms, we would like the radii of these monitoring spheres to be large enough so that each sphere covers not only the monitoring room at its center but also all its neighboring rooms, or perhaps, even beyond. However, all the problems that we consider in this thesis are with the condition that the monitoring spheres cover only up to the neighboring rooms of the monitoring rooms at the centers of these spheres. Therefore, in the graph-theoretic setup, such a monitoring sphere around a monitoring room is replaced by the set  $N_G[w] = \{v \in V(G) : d_G(v, w) \leq 1\}$  called the *closed neighborhood* of  $w$  in  $G$ , where  $w$  is a code vertex of  $G$ . Notice that the set  $N_G[w]$  has all vertices of  $G$  which are at distance at most 1 from the code vertex  $w$ .

We again digress a little for some further graph-theoretic definitions. A vertex subset  $S$  of a graph  $G$  is called a *dominating set* of  $G$  if any vertex of  $G$  not in  $S$  has at least one neighbor in  $S$ . The set of black vertices in Figure 1.2 is an example of a dominating set. Equivalently,  $S$  is a dominating set if  $N_G[v] \cap S \neq \emptyset$  for all  $v \in V(G)$ . A vertex  $v$  of  $G$  is said to *dominate* every vertex in  $N_G[v]$ . Another stronger notion of a dominating set is what is called a *total-dominating set* of a graph. The latter is defined as a vertex subset  $S$  of a graph  $G$  such that every vertex of  $G$  — and not just the ones not in  $S$  as in the definition of a dominating set — has a neighbor in  $S$ . The set of black vertices in Figure 1.2 serves also as an example of a total-dominating set in a graph. Now, define the set  $N_G(v) = N_G[v] \setminus \{v\}$  to be called the *open neighborhood*, or simply, the *neighborhood* of a vertex  $v$  in  $G$ . Then equivalently, a set  $S$  is a total-dominating set of a graph  $G$  if  $N_G(v) \cap S \neq \emptyset$  for all  $v \in V(G)$ . Moreover, a vertex  $v$  of  $G$  is said to *total-dominate* every vertex in  $N_G(v)$ . In general, this phenomenon of one vertex dominating (or total-dominating) another is also referred to as the property of *domination* in graphs.

Again coming back to our original discussion now in the graph-theoretic framework, we see that the whole concept of *monitoring* in real-life is therefore replaced by the concept of *domination* in graph theory. In addition, by Requirement (R2) above, since we want all rooms of the building to be monitored by the set of monitoring rooms, it implies in the graph-theoretic setting that, every vertex of  $G$  must be dominated by some code vertex of  $G$ . This further implies that to meet



Requirement (R2) of our monitoring system, a code of  $G$  must be a dominating set of  $G$ . With some additional requirements as we shall see later, sometimes the code of a graph  $G$  is required to be a total-dominating set as opposed to being just a dominating set of  $G$ . We shall discuss these requirements later.

One question that is natural to ask ourselves at this point is: *Do Requirements (R1) and (R2) imply Requirement (R3)? In other words, is requiring a code of a graph  $G$  to be a minimum dominating set of  $G$  enough to also meet the Requirement (R3) of our monitoring system?*

The answer to the above question is, in general, *No*. We explain our answer in more detail in Section 4.1 where we introduce the first kind of code that we shall consider in this chapter called the *locating-dominating code*.

Before we move on further, we remark here that, having established the above graph-theoretic language to describe on graphs the problem of monitoring rooms of a building, from here on, we choose to use the graph-theoretic vocabulary to describe all of what is to follow. Therefore, as much as we can, we replace in our writing terms like “building”, “room”, “adjacent rooms” and “monitoring” by the terms “graph”, “vertex”, “adjacent vertices” and “dominating”, respectively, and we also replace phrases like “intruder / detector in a room” by phrases like “intruder / detector at a vertex”.

One of the characteristics of the detectors that we have already introduced is that each detector, placed at a code vertex, say  $w$ , is able to monitor all vertices at distance at most one from  $w$ . We next impose a second characteristic on these detectors. Intuitively speaking, it would be our basic expectation, or at least not unreasonable to demand, that a code vertex  $w$  should be taken care of by the detector placed at  $w$  itself without any help from others. It is the non-code vertices without the detectors that need help from the code vertices to be monitored against any intrusion. Thus, the second characteristic that we attribute to the detectors is that each detector renders a code vertex to be *self-monitored*. In other words, whenever there is an intruder at a code vertex  $w$ , the detector placed at  $w$  is enough to signal so. We next layout a very simplified working principle of such detectors rendering code vertices self-monitored.

Each detector, placed at a code vertex, say  $w$ , has two indicator buttons — for example, two different light signals, say red and blue — which glow depending on which vertex in  $N_G[w]$  an intruder is present at. Whenever there is an intruder at the vertex  $w$ , both the red and the blue lights on the detector at  $w$  glow. On the other hand, whenever there is an intruder at any of the vertices in  $N_G(w)$ , it is *only* the blue light on the said detector at  $w$  that glows. Thus, if an intruder is present at a code vertex  $w$ , the only red light that glows is that of the detector at  $w$  (there may be several other blue lights at the same time depending on how many vertices in  $N_G[w]$  are code vertices). Therefore, by virtue of this unique red light, the system completely locates the position of the intruder, namely at  $w$ . This two-light working mechanism therefore partially meets the Requirement (R3) of our monitoring system the following way: if an intruder is present at a code vertex  $w$  of  $G$ , the system unambiguously points exactly to  $w$  to indicate so. Of course, this still leaves open the question as to how does the monitoring system uniquely identify an intruder at a non-code vertex and we shall answer that in Section 4.1. However, for now, it is clear that this two-light working mechanism of the detectors is a simplified model that can be used to endow each code vertex of a graph with the capacity to be self-monitored. To help in our writing, wherever necessary, let us refer to these detectors as *two-light detectors*.

In the next section, we develop the condition required to be imposed on a code of a graph in order for it to model a monitoring system satisfying Requirement (R3).

### 1.1.1 Locating-dominating codes

One of the most well-studied codes in the literature of identification problems is the *locating-dominating code*, or the *LD-code* for short. LD-codes was first introduced by Slater [186] in the 1980s under the name *locating-dominating set*.

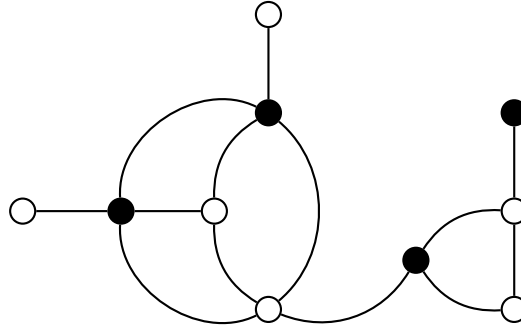


Figure 1.3: Example of an LD-code of a graph  $G$ . The black vertices are those that form an LD-code. In fact, the LD-code depicted in this figure is of the minimum possible cardinality for  $G$ . Thus, we have  $\gamma^{\text{LD}}(G) = 4$ .

We now address the question we asked before: Do Requirements (R1) and (R2) imply Requirement (R3)? To do so, we look at a monitoring system in a building using the two-light detectors. We have already seen that this two-light working mechanism partially meets Requirement (R3) in uniquely locating an intruder at any code vertex of a graph  $G$ . Thus, what remains is to establish the condition on a code of  $G$  such that the monitoring system meets Requirement (R3) fully, that is, it also uniquely identifies an intruder at a non-code vertex of  $G$ . Let  $S$  be the code in a graph  $G$  that models a monitoring system using two-light detectors. Now, if an intruder is present at a non-code vertex, say  $v$ , then the system produces only blue lights and, specifically, from those detectors at vertices in  $N_G(v) \cap S$ . Let there exist another non-code vertex  $u$  of  $G$  dominated by the same set of code vertices that dominate  $v$ , that is,  $N_G(u) \cap S = N_G(v) \cap S$ . This implies that the same set of detectors monitor both the vertices  $u$  and  $v$ . Therefore, in case an intruder is present at either  $u$  or  $v$ , the (only) blue lights emanating from the system are those of the detectors at the vertices in  $N_G(v) \cap S$ . This implies that the intruder could be in either of the rooms  $u$  and  $v$  simultaneously. Hence, for the code  $S$  of  $G$  to model a monitoring system meeting Requirement (R3), it must be such that  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G) \setminus S$ . This phenomenon is, in general, called the *separation* of vertices and the two vertices  $u$  and  $v$  are said to be *separated* by the code  $S$ . Thus, a code  $S$  of a graph  $G$  must not only be a dominating set of  $G$  but must also be able to separate pairs of distinct vertices of  $G$ . Hence, in general, a code of  $G$  can be of order larger than a minimum dominating set of  $G$ . This answers the above question negatively.

Our above analysis therefore motivates the following code of a graph.

**Definition 1.1** (Locating-dominating code (LD-code)). Given a graph  $G$ , a *locating-dominating code* of  $G$ , or an *LD-code* for short, is a vertex subset  $S$  of  $G$  with the following two properties.

- (1) (property of *domination*)  $S$  is a dominating set of  $G$ , that is,  $N_G[v] \cap S \neq \emptyset$  for each vertex  $v \in V(G)$ .
- (2) (property of *location*)  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G) \setminus S$ .

See Figure 1.3 for an example of an LD-code of a graph. Another equivalent way to think about Condition (2) of Definition 1.1 is to say that, for each vertex  $v \in V(G) \setminus S$ , the set  $N_G(v) \cap S$  is *unique*. In other words, an LD-code is a set such that any non-code vertex is dominated by a unique set of code vertices. As already addressed in our analysis before, the points (1) and (2) of Definition 1.1 ensure that the monitoring system modelled by an LD-code satisfies Requirements (R2) and (R3). Hence, an LD-code of the minimum possible cardinality is a code that models a monitoring system meeting Requirements (R1) – (R3). We denote the minimum possible cardinality of all LD-codes of a graph  $G$  by the symbol  $\gamma^{\text{LD}}(G)$  and call it the *LD-number* of  $G$ . The LD-code depicted in Figure 1.3 is, in fact, an LD-code of the minimum possible cardinality of the graph.

## 1.1.2 Identifying codes

While the two-light detectors can devise a monitoring system meeting Requirements (R1) – (R3), there may be some practical challenges that one may have to consider with regards to the functioning, longevity and susceptibility to damages of these detectors. For example, some of them may become faulty over time, or the proper reporting from these devices may be hampered due to certain external factors like interference from other detectors. One of the faults that has been considered in the literature of identification problems is the following.

**Fault Type 1.** Certain code vertices do not remain self-monitored any more — that is, the red light on some of the two-light detectors stops working.

Let us start with an LD-code  $S$  of a graph  $G$  with the understanding that detectors at some of the code vertices may undergo fault type 1. In the worst-case scenario, let us assume that the red lights on all the detectors have stopped working. Then any signal to indicate the presence of an intruder at any vertex  $v$  of  $G$  must come from *only* the blue lights on the detectors at the vertices in  $N_G[v] \cap S$ . This does not hamper the separation by the code  $S$  of one non-code vertex from another, since for a non-code vertex  $v$ , we have  $N_G[v] \cap S = N_G(v) \cap S$ . Therefore, with  $S$  being an LD-set, we have  $N_G[u] \cap S \neq N_G[v] \cap S$  for all distinct non-code vertices  $u, v \in V(G)$ . However, there may be a code vertex  $w$  which may not be separated by  $S$  from another vertex, say  $v$ , if  $N_G[w] \cap S = N_G[v] \cap S$ . In such a case, if  $N_G[w] \neq N_G[v]$ , then there exists a vertex  $x \in N_G[w] \triangle N_G[v]$ , where the set  $A \triangle B = (A \setminus B) \cup (B \setminus A)$  is called the *symmetric difference* between any two sets  $A$  and  $B$ . Therefore, augmenting  $S$  to  $S \cup \{x\}$  resolves the non-separation of the vertices  $w$  and  $v$  by the previous code. Thus, if the graph  $G$  is such that  $N_G[u] \neq N_G[v]$  for all distinct vertices  $u, v \in V(G)$ , then, the initial LD-set  $S$  can be augmented with further vertices such that  $N_G[u] \cap S \neq N_G[v] \cap S$  for all distinct vertices  $u, v \in V(G)$ . The new code  $S$  of  $G$  therefore models a monitoring system that can withstand its detectors undergoing fault type 1 by producing a set of blue lights — namely, those on the detectors at  $N_G[v] \cap S$  — completely unique to the vertex  $v \in V(G)$  in case an intruder is present at  $v$ .

One crucial observation from the analysis in the previous paragraph is that in the absence of the condition  $N_G[u] \neq N_G[v]$  for all distinct vertices  $u, v \in V(G)$ , we could not construct a code which can withstand fault type 1. Any two distinct vertices  $u$  and  $v$  of any graph  $G$  are called *closed twins* in  $G$  if  $N_G[u] = N_G[v]$ . Moreover, a graph  $G$  with the property that  $N_G[u] \neq N_G[v]$  for all distinct  $u, v \in V(G)$  is called *closed-twin-free*. This motivates us to define the following code of a graph.

**Definition 1.2** (Identifying code (ID-code)). Given any closed-twin-free graph  $G$ , a vertex subset  $S$  of  $G$  is called an *identifying code* (or an *ID-code* for short) of  $G$  if  $S$  has the following two properties.

- (1) (property of *domination*)  $S$  is a dominating set of  $G$ .
- (2) (property of *closed separation*)  $N_G[u] \cap S \neq N_G[v] \cap S$  for all distinct vertices  $u, v \in V(G)$ .

See Figure 1.4 for an example of an ID-code of a graph. Another equivalent way to think about Condition (2) of Definition 1.2 is to say that, for each vertex  $v \in V(G)$ , the set  $N_G[v] \cap S$  is *unique*. In other words, an ID-code is a set such that any vertex is dominated by a unique set of code vertices.

Identifying codes were first introduced by Karpovsky et al. [149] in 1998. They have also been called *differentiating dominating sets* (see for example, [113]). As is evident, an ID-code of a graph models a monitoring system that withstands its detectors undergoing fault type 1. As before, we are interested in ID-codes of any closed-twin-free graph  $G$  of the minimum possible cardinality denoted by the symbol  $\gamma^{\text{ID}}(G)$  and is called the *ID-number* of  $G$ . The ID-code depicted in Figure 1.4 is, in fact, an ID-code of the minimum possible cardinality of the graph.

One interesting aspect to note from Definition 1.2 is that an ID-code is also an LD-code. This is consistent with our analysis above where we augmented an existing LD-code with further vertices to turn it into an ID-code that tackles fault type 1.

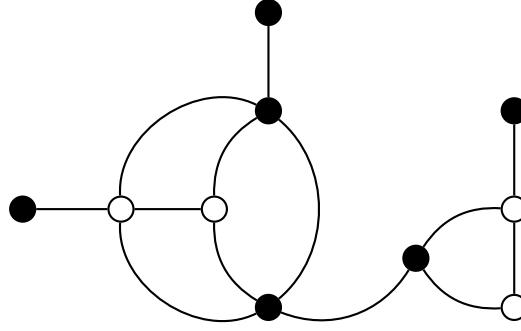


Figure 1.4: Example of an ID-code of a graph  $G$ . The black vertices are those in an ID-code of the minimum cardinality possible for  $G$ . Therefore, we have  $\gamma^{\text{ID}}(G) = 6$ .

### 1.1.3 Open-separating total-dominating codes

Another well-studied code in the literature arises out of considering the following practical challenge that a monitoring system may encounter.

**Fault Type 2.** The intruder at a code vertex  $w$  of a graph is a fire or a saboteur which may completely disable or destroy the detector installed at  $w$ . In this case, both the red and blue buttons on a particular detector are rendered dysfunctional.

Let us again start with an LD-code  $S$  of a graph  $G$  with the understanding that a detector at some code vertex, say  $w$ , has undergone fault type 2. This is equivalent to modifying the set  $S$  to  $S \setminus \{w\}$ . This renders all vertices of  $G$  *solely* dominated (monitored) by  $w$  to become undominated. Moreover, if there is a pair of distinct non-code vertices  $u, v \in V(G) \setminus S$  that were separated *solely* by  $w \in S$ , that is,  $(N_G(u) \cap S) \triangle (N_G(v) \cap S) = \{w\}$ , then the new code  $S \setminus \{w\}$  is not able to separate these two vertices any more. Hence, if the saboteur at  $w$ , after having disabled the detector there, spreads to any other vertex, it may not be able to locate them any further at any particular vertex — or at least with probability 1. The code which we shall formally introduce next is called *open-separating total-dominating code*, or *OTD-code* for short, has so far been studied the most in the literature with regards to modeling a monitoring system undergoing fault type 2. However, as has been pointed out, a monitoring system modeled by an OTD-code requires that the saboteur be seized at the code vertex  $w$  itself, whose detector they disable, before they spread to other vertices of  $G$ . Some efforts have been made more recently to devise other kinds of codes of graphs — which we shall only mention in passing in Section 10.2.1 — that are successful in tackling the situation even when an intruder spreads to other vertices.

To overcome a challenge of fault type 2, it is important that a code vertex  $w$  is not dependent only on itself for the detection of an intruder at  $w$ . In other words, the monitoring system must be such that each code vertex  $w$  is itself dominated by a unique set of code vertices of  $G$  other than  $w$ . This implies that  $N_G(w) \cap S \neq \emptyset$  and, moreover,  $N_G(w) \cap S$  must be unique for each  $w \in S$ . Similarly, as before, for any non-code vertex  $v$  of  $G$ , since  $v$  is monitored only by the detectors at the vertices in  $N_G(v) \cap S$ , the latter set must also be both non-empty and unique for each non-code vertex  $v$ . Therefore, the code  $S$  must be such that  $N_G(u) \cap S \neq \emptyset$  and  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G)$ . However, the last condition requires that the graph  $G$  is *open-twin-free*, that is,  $N_G(u) \neq N_G(v)$  for all distinct vertices  $u, v \in V(G)$ . Moreover, any two distinct vertices  $u, v \in V(G)$  with  $N_G(u) = N_G(v)$  are called *open-twins* in  $G$ . We also notice from our analysis that we must also have  $N_G(v) \cap S \neq \emptyset$  for all  $v \in V(G)$ . This is ensured only if  $G$  is *isolate-free*, that is, each vertex of  $G$  has some edge incident with it.

**Definition 1.3** (Open-separating total-dominating code (OTD-code)). Given an open-twin-free and isolate-free graph  $G$ , an *open-separating total-dominating code* of  $G$ , or an *OTD-code* for short, is a vertex subset  $S$  of  $G$  with the following two properties.

- (1) (property of *total domination*)  $S$  is a total-dominating set of  $G$ .
- (2) (property of *open separation*)  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G)$ .

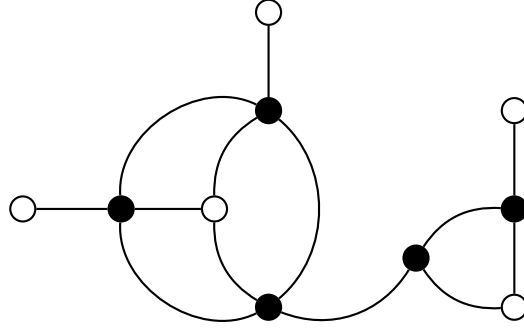


Figure 1.5: Example of an OTD-code of a graph  $G$ . The black vertices are those that form an OTD-code. Moreover, this code is of the minimum possible cardinality for  $G$ . Therefore, we have  $\gamma^{\text{OTD}}(G) = 5$ .

See Figure 1.5 for an example of an OTD-code of a graph. The concept of an OTD-code was first introduced in a more general setting by Honkala et al. in [133] under the concepts of *strongly* ( $t, \leq l$ )-*identifying codes*, and then later by Seo and Slater [190] under the name of *open neighborhood locating-dominating set*. Analogous to the previous codes, another equivalent way to think about Condition (2) of Definition 1.3 is to say that the code  $S$  must be such that for each vertex  $v \in V(G)$ , the set  $N_G(v) \cap S$  is *unique*. In other words, an OTD-code is a set such that any vertex is total-dominated by a unique set of its code vertices. Finally, as before, we are interested in finding the minimum value, denoted by  $\gamma^{\text{OTD}}(G)$  and called the *OTD-number* of  $G$ , among the cardinalities of all OTD-codes of a graph  $G$ . As before, the OTD-code depicted in Figure 1.5 is, in fact, a minimum-ordered OTD-code of the graph.

### 1.1.4 Full-separating total-dominating codes

In the case that we would like to build a monitoring system that resist both fault types 1 and 2, the corresponding code  $S$  in a graph  $G$  must be both an ID-code and an OTD-code of  $G$ . Then, from our previous analyses of ID-codes and OTD-codes, it follows that the graph  $G$  must be isolate-free and both closed-twin-free and open-twin-free. We call such a graph that is both closed- and open-twin-free to be *twin-free*.

**Definition 1.4** (Full-separating total-dominating code (FTD-code)). Given a twin-free and isolate-free graph  $G$ , a *full-separating total-dominating code*, or an *FTD-code* for short, of  $G$ , first introduced by Chakraborty (the author) and Wagler [53], is a vertex subset  $S$  of  $G$  with the following two properties.

- (1) (property of *total domination*)  $S$  is a total-dominating set of  $G$ .
- (2) (property of *full separation*)  $N_G[u] \cap S \neq N_G[v] \cap S$  and  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G)$ .

See Figure 1.6 for an example of an FTD-code of a graph. Analogous to the previous codes, an equivalent way to think about Condition (2) of Definition 1.2 is to say that the code  $S$  must be such that for each vertex  $v \in V(G)$ , the set  $N_G[v] \cap S$  is *unique* among all sets in the set  $\{N_G[u] : u \in V(G)\}$  and the set  $N_G(v) \cap S$  is *unique* among all sets in the set  $\{N_G(u) : u \in V(G)\}$ . In other words, an FTD-code is a set such that any vertex is both dominated and total-dominated by a unique set of its code vertices. We denote the minimum value among the cardinalities of all FTD-codes of a

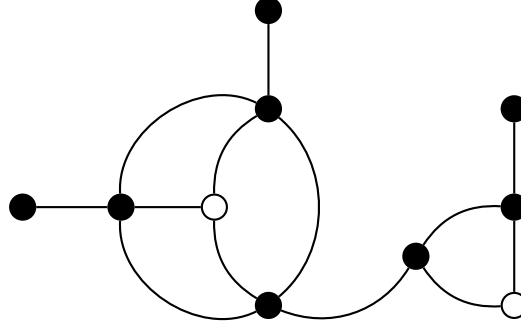


Figure 1.6: Example of an FTD-code of a graph  $G$ . The black vertices form FTD-code of  $G$  which is also of the minimum possible cardinality. Thus, we have  $\gamma^{\text{FTD}}(G) = 8$ .

graph  $G$  by  $\gamma^{\text{FTD}}(G)$  and call it the *FTD-number* of  $G$ . In this case too, the black vertices in Figure constitute a minimum-ordered FTD-code of the graph.

As can be noted from Figures 1.3 – 1.6, for the same graph  $G$ , all four values of  $\gamma^{\text{LD}}(G)$ ,  $\gamma^{\text{ID}}(G)$ ,  $\gamma^{\text{OTD}}(G)$  and  $\gamma^{\text{FTD}}(G)$  differ from one another. This motivates the study of each of the codes individually both from a theoretical and a practical point of view.

### 1.1.5 More codes based on domination

From a mathematical point of view, each of the four codes introduced so far is also interesting to study by replacing its property of domination (respectively, total domination) by the property of total domination (respectively, domination). As a result, the following four more identification problems on graphs naturally present themselves in the literature.

**Definition 1.5** (Locating-total-dominating code (LTD-code)). Given an isolate-free graph  $G$ , a *locating-total-dominating code* of  $G$ , or *LTD-code* for short, first introduced by Haynes et al. [123] under the name *locating-total dominating set*, is a vertex subset  $S$  of  $G$  with the following two properties.

- (1) (property of *total domination*)  $S$  is a total-dominating set of  $G$ .
- (2) (property of *location*)  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G) \setminus S$ .

The minimum value among the cardinalities of all LTD-codes of a graph  $G$  is denoted by  $\gamma^{\text{LTD}}(G)$  and is called the *LTD-number* of  $G$ . See Figure 1.7a for an example of an LTD-code of a graph.

**Definition 1.6** (Identifying total-dominating code (ITD-code)). Given a closed-twin-free and isolate-free graph  $G$ , an *identifying total-dominating code* of  $G$ , or *ITD-code* for short, first introduced by Haynes et al. [123] under the name *differentiating-total dominating set*, is a vertex subset  $S$  of  $G$  such that it has the following two properties.

- (1) (property of *total domination*)  $S$  is a total-dominating set of  $G$ .
- (2) (property of *closed separation*)  $N_G[u] \cap S \neq N_G[v] \cap S$  for all distinct vertices  $u, v \in V(G)$ .

The minimum value among the cardinalities of all ITD-codes of a graph  $G$  is denoted by  $\gamma^{\text{ITD}}(G)$  and is called the *ITD-number* of  $G$ . See Figure 1.7b for an example of an ITD-code of a graph.

**Definition 1.7** (Open-separating dominating code (OD-code)). Given an open-twin-free graph  $G$ , an *open-separating dominating code* of  $G$ , or *OD-code* for short, first introduced by Chakraborty (the author) and Wagler [52], is a vertex subset  $S$  of  $G$  with the following two properties.

- (1) (property of *domination*)  $S$  is a dominating set of  $G$ .

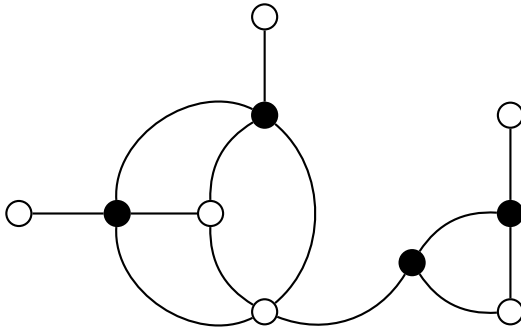
- (2) (property of *open separation*)  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G)$ .

The minimum value among the cardinalities of all OD-codes of a graph  $G$  is denoted by  $\gamma^{\text{OD}}(G)$  and is called the *OD-number* of  $G$ . See Figure 1.7c for an example of an OD-code of a graph.

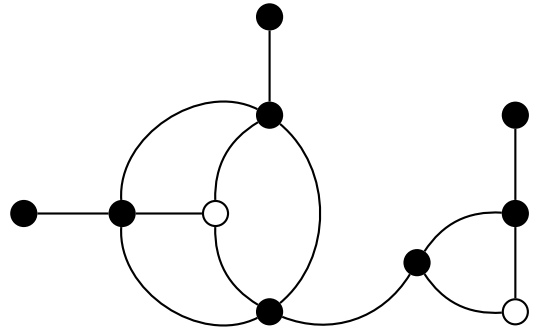
**Definition 1.8** (Full-separating dominating code (FD-code)). Given a twin-free graph  $G$ , a *full-separating dominating code* of  $G$ , or *FD-code* for short, first introduced by Chakraborty (the author) and Wagler [53], is a vertex subset  $S$  of  $G$  with the following two properties.

- (1) (property of *domination*)  $S$  is a dominating set of  $G$ .  
 (2) (property of *full separation*)  $N_G[u] \cap S \neq N_G[v] \cap S$  and  $N_G(u) \cap S \neq N_G(v) \cap S$  for all distinct vertices  $u, v \in V(G)$ .

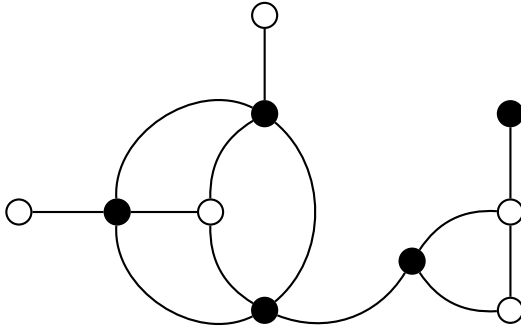
The minimum value among the cardinalities of all FD-codes of a graph  $G$  is denoted by  $\gamma^{\text{FD}}(G)$  and is called the *FD-number* of  $G$ . See Figure 1.7d for an example of an FD-code of a graph.



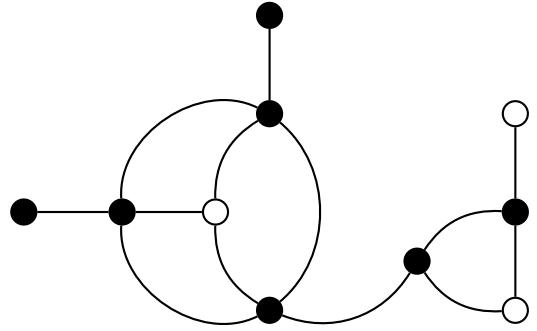
(a) Example of an LTD-code of a graph  $G$ . The black vertices form an LTD-code of the minimum possible cardinality for the graph. Thus,  $\gamma^{\text{LTD}}(G) = 4$ . Therefore, in this case, we have  $\gamma^{\text{LTD}}(G) = \gamma^{\text{LD}}(G)$ . However, this may not be the case in general. Moreover, the LD-code in Figure 1.3 is not the same as that for LTD-code here.



(b) Example of an ITD-code of a graph  $G$ . The black vertices are those that form an ITD-code of the minimum possible cardinality of the graph. Thus,  $\gamma^{\text{ITD}}(G) = 8$ . Therefore, in this case, we see that  $\gamma^{\text{ID}}(G) < \gamma^{\text{ITD}}(G) = \gamma^{\text{FTD}}(G)$ .



(c) Example of an OD-code of a graph  $G$ . The black vertices are those that form an OD-code. Moreover, this OD-code is of the minimum possible cardinality of the graph. Thus,  $\gamma^{\text{OTD}}(G) = 5$ . Therefore, in this case, we have  $\gamma^{\text{OD}}(G) = \gamma^{\text{OTD}}(G)$  but this is not true in general. Moreover, the OD-set in this figure is not an OTD-set of  $G$ .



(d) Example of an FD-code of a graph  $G$ . The black vertices form an FD-code of  $G$  that is of the minimum possible cardinality of the graph. Thus,  $\gamma^{\text{FD}}(G) = 7$ . Therefore, in this case, we have  $\gamma^{\text{FD}}(G) < \gamma^{\text{FTD}}(G)$ .

Again, it is evident from the Figures 1.7a to 1.7d that the respective code numbers vary for the same graph. Thus, we feel that all the eight codes that we have introduced so far, warrant a study both individually and in comparison to one another; and this is what we do in this thesis. From now on, as a matter of abbreviation, we define  $\text{CODES} = \{\text{LD}, \text{LTD}, \text{ID}, \text{ITD}, \text{OD}, \text{OTD}, \text{FD}, \text{FTD}\}$ . As we have lightly pointed out in this chapter (and will do so more rigorously in Chapter 2), an X-code of a given graph may or may not exist. However, if an X-code of a graph  $G$  exists, then we shall call  $G$  to be *X-admissible*. Moreover, recapitulating the notations and terminologies introduced so far, the minimum cardinality of an X-code in an X-admissible graph  $G$  is called the X-number of  $G$  and is denoted by  $\gamma^X(G)$ .

## 1.2 Other applications of codes

The room monitoring system in a building is a prototype of what identification problems can be applied to. As the reader must have already conceptualized by now, such problems can be useful in any real-life network wherever one has the need to distinguish one element from another. We present here a small sample of some other prominent applications of identification problems.

- (1) Allotting unique identification numbers — also called the *naming problem* [156] — to each entity in a network like the social media.
- (2) Monitoring and fault-detection in a network of multiprocessors [179, 149].
- (3) Minimizing the cost of *tests* [166, 30] which encompasses a variety of studies — all based on the idea of distinguishing objects — like
  - (a) pattern recognition [78, 168],
  - (b) biological identification [174, 200, 201],
  - (c) fault diagnosis and analysis [20, 38, 161]
- (4) Fire protection studies of nuclear power plants [135, 136, 137].
- (5) Monitoring systems of building complexes based on fire alarms or motion sensors [180, 190, 198].
- (6) *Routing problems* [156] used for transmitting messages from one node to another through some special node called *routers* usually spread out as a dominating set across the network.
- (7) Comparison and analytical studies of secondary RNA molecules modeled as graphs [126].
- (8) To provide upper bounds on *definability of graphs* [151], a tool used to describe the complexity of the graph in First Order logic.
- (9) Identification problems are used in coding theory [60] and, in fact, it is from this domain that the word “code” has come to be associated with this topic.
- (10) Several other related areas like bioinformatics [22], coin-weighing problems [189], graph isomorphism [15], games [64] and machine learning [61].

## 1.3 Organization of the thesis

The contents of this thesis are categorized primarily into three parts. The first three chapters look at the problems considered in this thesis from the most general point of view.

Chapter 1 is aimed at introducing the different problems that have been considered in this work. It also deals with the various applications and motivations to study these identification problems.

Chapter 2 is where most of the definitions, notations and terminologies are introduced. It is also where the rigorous framework (both mathematical and graph-theoretic) is laid out for these concepts to be used several times over the next chapters to come. The second half of this chapter also



provides a brief literature survey — of both combinatorial and algorithmic results — of the problems considered in this thesis.

Chapter 3 deals with laying out some more mathematical foundations — and more specifically, alternative reformulations in terms of hypergraphs — of studying identification problems. This chapter also provides a comprehensive study of the pairwise comparisons of all the eight code-numbers considered in this work. Moreover, the chapter also looks at the most general upper and lower bounds that can be provided to these code numbers of graphs in terms of the order of the graph.

Chapters 4 to 7 deal with the combinatorial aspects — mostly dealing with bounds on code numbers — of the codes we study in this thesis. More specifically, each chapter in this range is dedicated to the focused study of each of the four types of separation properties that we study in this thesis. Chapter 4 is dedicated to the study of location as a separation property on several graph families, Chapter 5 is aimed at studying identifying codes on some graph families, Chapter 6 looks at the property of open separation on graph families and lastly, Chapter 7 is dedicated to the study of the property of full separation in several graph families.

Chapters 8 and 9 are dedicated to the study of the algorithmic and complexity aspects of the identification problems considered in this thesis. More precisely, Chapter 8 addresses the study of the computational complexities of the problems of finding minimum OD-, FD- and FTD-codes of graphs; and some other related decision problems. Thereafter, Chapter 9 is dedicated to the study of finding minimum LD-codes of graphs with some additional “relatively small” parameters given as input to the problem.

Finally, in Chapter 10, we summarize our work as a whole and provide perspectives of the problems studied in this thesis in the context of other related and ongoing identification problems considered in this research area. This also paves the way for other potential problems and questions that may be considered as future work.

## 1.4 Other work carried out during PhD

Apart from the study presented in this thesis, the author has also been involved in the other research activities, teaching, organizing committee of academic events and co-supervising activities.

In terms of other research activities undertaken during the PhD, the author has contributed to the following research. In [50], the algorithmic and complexity aspects of the graph modification problem of *edge contraction* was studied. In [49], a generalized version of graph coloring called the *radio  $k$ -coloring* was studied. In [26], the problems of LD-codes, LTD-codes and OTD-codes were studied for the *Mycielski construction* of a given graph. Finally, in [46], two other kinds of identification problems called *locating coloring* and *neighbor-locating coloring*, using graph coloring rather dominating sets, have been studied.

The author has also carried out some teaching activities in all three years of the PhD. The job role of this activity during this period includes “Travaux Dirigés” (TD — class tutorials) and “Travaux Pratiques” (TP — practical sessions) of a course on Optimization (Methodes d’Optimization) for the second year students of Bachelor Universitaire de Technologie (BUT) Informatique at l’Institut Universitaire de Technologie (IUT) at Université Clermont Auvergne, Clermont-Ferrand, France.

The author was also one of the members of the local organizing committee of the 41st International Symposium on Theoretical Aspects of Computer Science (STACS 2024) held at Université Clermont Auvergne, Clermont-Ferrand, France. Finally, the author has also carried out co-supervision (along with Prof. Foucaud) of an intern from the Chennai Mathematical Institute (CMI), India on the topic of local identification problems in graphs.

# Chapter 2

## Preliminaries and overview

In this chapter we first establish the basic mathematical and notational foundations that we shall be using for the rest of the thesis. To that end, some of the definitions already introduced in Chapter 1 will be reintroduced in this chapter mostly in a more rigorous mathematical setting. We also formally introduce several familiar graph families and provide a brief literature survey of the problems at hand, both from the structural and algorithmic point of view.

### 2.1 Notations and terminologies

In this section we layout the basic mathematical and graph-theoretic notations that we use throughout the thesis. Most of our notations are standard and commonly used within the community. We also refer the reader to the book on graph theory by West [199] for all standard notations and terminologies.

#### 2.1.1 General mathematics

Given any set  $S$ , the number of elements in  $S$  is denoted by  $|S|$  and is called the *order* or the *cardinality* of the set  $S$ . Also, we let  $A - B = A \setminus B = \{a \in A : a \notin B\}$ . Moreover, if  $B = \{b\}$  is a singleton set, then for notational convenience, we often write  $A + b$  and  $A - b$  instead of  $A \cup \{b\}$  and  $A \setminus \{b\}$ , respectively. Let  $\mathbb{Z}$  denote the set  $\{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$  and be called the *set of integers*. Any element of  $\mathbb{Z}$  is called an *integer*. Moreover, let  $\mathbb{N}$  denote the set  $\{0, 1, 2, 3, \dots\}$  and be called the *set of non-negative integers*. Any element of  $\mathbb{N}$  is called a *non-negative integer*. In addition, the set  $\mathbb{N}^* = \mathbb{N} - 0$  is called the *set of positive integers* and any element of  $\mathbb{N}^*$  is called a *positive integer*. For any positive integer  $k$ , the notation  $[k]$  denotes the set  $\{1, 2, \dots, k\}$ . Moreover, for any two (not necessarily positive) integers  $k$  and  $l$  such that  $k < l$ , the notation  $[k, l]$  denotes the set  $\{k, k + 1, k + 2, \dots, l\}$ .

The set of all *real numbers* is denoted by  $\mathbb{R}$ . Sometimes  $\mathbb{R}$  is also referred to as the *real number line*, as it can be represented by a line on the plane. Let  $a$  and  $b$  be two real numbers such that  $a < b$ . An *open interval* (between  $a$  and  $b$ ) on the real number line is defined as the set  $\{x \in \mathbb{R} : a < x < b\}$  and is denoted by  $(a, b)$ . Similarly, a *closed interval* (between  $a$  and  $b$ ) on the real number line is defined as the set  $\{x \in \mathbb{R} : a \leq x \leq b\}$  and is denoted by  $[a, b]$ . Moreover, the notations  $(a, b]$  and  $[a, b)$  also denote the sets  $\{x \in \mathbb{R} : a < x \leq b\}$  and  $\{x \in \mathbb{R} : a \leq x < b\}$ , respectively. All the above sets are in general also referred to as an *interval* on the real number line.

By the notation  $A \triangle B$ , we denote the set  $(A \setminus B) \cup (B \setminus A)$  and call it the *symmetric difference* between  $A$  and  $B$ . The *Cartesian product* of two sets  $A$  and  $B$  is the set  $\{(a, b) : a \in A \text{ and } b \in B\}$  and is denoted by  $A \times B$ . A *relation* on a set  $S$  is a subset of the Cartesian product  $S \times S$ . A relation  $R$  on  $S$  is called *reflexive* if  $(s, s) \in R$  for all  $s \in S$ .  $R$  is called *symmetric* if  $(s, s') \in R$  whenever

$(s', s) \in R$ , where  $s, s' \in S$ . Moreover,  $R$  is called *transitive* if, for any three elements  $s, s', s'' \in S$ ,  $(s, s') \in R$  and  $(s', s'') \in R$  imply that  $(s, s'') \in R$ . A relation  $R$  on a set  $S$  is called an *equivalence relation* on  $S$  if  $R$  is reflexive, symmetric and transitive. In addition, a relation  $R$  on a set  $S$  is called *antisymmetric* if for any two  $s, s' \in S$ , the condition  $(s, s') \in R$  and  $(s', s) \in R$  implies that  $s = s'$ . Furthermore, a set  $S$  is called *partially ordered* if there exists a relation  $R$  on  $S$  such that  $R$  is transitive, antisymmetric and transitive.

Given any set  $S$ , it can be verified that  $S \times S$  is an equivalence relation on  $S$ . Hence, an equivalence relation on any set always exists. Let  $S$  be a set, let  $R$  be an equivalence relation on  $S$  and let  $s \in S$ . Then define a set  $[s]_R \subseteq S$  by  $[s]_R = \{s' \in S : (s, s') \in R\}$ . Since  $R$  is reflexive, by the fact that  $(s, s) \in R$ , we have  $s \in [s]_R$  and hence,  $[s]_R \neq \emptyset$  for all  $s \in S$ . The set  $[s]_R$  is called an *equivalence class* of  $R$  and the element  $s$  is called a *representative* of the class  $[s]_R$ .

A *partition* of a set  $S$  is a set  $\mathcal{P} = \{S_1, S_2, \dots, S_k\}$ , for some positive integer  $k$ , such that  $S_i \subseteq S$  for each  $i \in [k]$ ,  $S_i \cap S_j = \emptyset$  for all  $i, j \in [k]$  with  $i \neq j$  and  $S_1 \cup S_2 \cup \dots \cup S_k = S$ . Each  $S_i$  is called a *part* of the partition  $\mathcal{P}$ . Let  $R$  be an equivalence relation on a set  $S$ . Then, it can be verified that the set of all equivalence classes of  $R$  is a partition of  $S$ . On the other hand, given a partition  $\mathcal{P} = \{S_1, S_2, \dots, S_k\}$  of a set  $S$ , it can be verified that the set  $R = \{(s, s') : s, s' \in S_i \text{ for some } i \in [k]\}$  defines an equivalence relation on  $S$ . Hence, we have the following remark.

**Remark 2.1.** *Given a set  $S$ , there is a bijection between the set of all partitions on  $S$  and the set of all equivalence relations on  $S$ .*

Given two sets  $A$  and  $B$ , a *function* from  $A$  to  $B$ , denoted as  $f : A \rightarrow B$ , implies that, to each element  $a$  of  $A$ , an element  $f(a)$  from  $B$  is assigned. The element  $f(a) \in B$  assigned to  $a \in A$  is called the *image of  $a$*  under the function  $f$ . If  $f(a) \neq f(a')$  for all pairs  $a, a' \in A$ , then  $f$  is called an *injective function* or an *injection* or a *one-to-one function*. If for every  $b \in B$ , there exists an element  $a \in A$  such that  $f(a) = b$ , then  $f$  is called a *surjective function* or a *surjection* or an *onto function*. Moreover, if  $f$  is both injective and surjective, then  $f$  is called a *bijective function* or a *bijection*.

Given a function  $f : A \rightarrow B$  and a subset  $C \subseteq B$ , we define the set  $f^{-1}(C) = \{a \in A : f(a) \in C\}$  and call it the *inverse image* of the set  $C$  under the function  $f$ . Sometimes, when  $C = \{c\}$  is a singleton, we simply write  $f^{-1}(c)$  instead of  $f^{-1}(\{c\})$  for notational convenience. Moreover, when a function  $f : A \rightarrow B$  is either well-understood or is not important to the context, it may be denoted in symbol by  $A \rightarrow B$ . If  $A$  is a subset of a set  $B$ , then  $A \hookrightarrow B$  denotes the *inclusion function* whereby, the image of each  $a \in A$  under the function is  $a \in B$ .

Given two functions  $f : \mathbb{R} \rightarrow \mathbb{R}$  and  $g : \mathbb{R} \rightarrow \mathbb{R}$  such that  $g(x) > 0$  for all  $x \in \mathbb{R}$ , we denote  $f(x) = \mathcal{O}(g(x))$  if there exists an  $x_0 \in \mathbb{R}$  and a positive constant  $c \in \mathbb{R}$  such that  $|f(x)| \leq c \cdot g(x)$  for all  $x \geq x_0$ . If there exists a positive constant  $c \in \mathbb{R}$  and an  $x_0 \in \mathbb{R}$  such that  $|f(x)| \geq c \cdot g(x)$  for all  $x \geq x_0$ , then we denote it by  $f(x) = \Omega(g(x))$ . Moreover, if  $f(x) = \mathcal{O}(g(x))$  and  $f(x) = \Omega(g(x))$ , then we denote it by  $f(x) = \Theta(g(x))$ . On the other hand, we denote  $f(x) = o(g(x))$  if for any given positive constant  $c \in \mathbb{R}$ , there exists an  $x_0 \in \mathbb{R}$  such that  $|f| \leq c \cdot g(x)$  for all  $x \geq x_0$ . Moreover, we denote it by  $g(x) = \omega(g(x))$  when for every positive constant  $c \in \mathbb{R}$ , there exists an  $x_0 \in \mathbb{R}$  such that  $|f(x)| \geq c \cdot g(x)$ . Similarly, if  $f : \mathbb{N} \rightarrow \mathbb{N}$  and  $g : \mathbb{N} \rightarrow \mathbb{N}$ , then all of the above notations are used by simply replacing  $x$  with  $n$ . In other words, in each of the respective cases, we write  $f(n) = \mathcal{O}(g(n))$ ,  $f(n) = \Omega(g(n))$ ,  $f(n) = o(g(n))$  and  $f(n) = \omega(g(n))$ . All the above notations are known as *asymptotic notations*.

## 2.1.2 Graph theory

Let  $V(G)$  be any set and, for any integer  $k \geq 1$ , let  $A_1(G), A_2(G), \dots, A_k(G)$  be relations on  $V(G)$  such that  $A_1(G) \subseteq A_2(G) \subseteq \dots \subseteq A_k(G)$ . Then, the  $(k+1)$ -tuple  $G = (V(G), A_1(G), A_2(G), \dots, A_k(G))$  is called a *directed multigraph*. The set  $V(G)$  is called the *vertex set* of  $G$  and the set  $A_k(G)$  is called the *arc set* of  $G$ . Moreover, each element of the set  $V(G)$  is called a *vertex* of  $G$  and each element of  $A_k(G)$  is called an *arc* of  $G$ . For notational convenience, we denote any arc of  $G$  by  $uv$

rather than  $(u, v)$ , where  $u, v \in V(G)$ . If  $uu \in A_k(G)$ , then the arc  $uu$  is called a *loop* at the vertex  $u$ . A vertex  $u$  is called a *source* if  $vu \notin A_k(G)$  for all  $v \in V(G)$ ; and  $u$  called a *sink* if  $uv \notin A_k(G)$  for all  $v \in V(G)$ . The cardinality of the set  $V(G)$  is called the *order* of the graph  $G$ . Moreover, a graph of order  $n$  is said to be *on  $n$  vertices*. Any subset of  $V(G)$  is called a *vertex subset* of  $G$  and any subset of  $A_k(G)$  is called an *arc subset* of  $G$ . See Figure 2.1a for an example of a directed multigraph.

Let  $G = (V(G), A_1(G), A_2(G), \dots, A_k(G))$  be a directed multigraph. In the case that  $k = 1$ ,  $G$  is called a *directed graph* (or a *digraph* for short). If  $G$  is a digraph, then the set  $A_1(G)$  is simply denoted by  $A(G)$ . See Figure 2.1b for an example of a directed graph. A digraph  $G$  is called an *oriented graph* if  $|\{uv, vu\} \cap A(G)| \leq 1$  for all  $u, v \in V(G)$ . See Figures 2.1c and 2.1d for examples of oriented graphs. If the arc set of a digraph  $G$  is a symmetric relation on its vertex set, then  $G$  is called an *undirected graph*. The arc set of an undirected graph  $G$  is called an *edge set* of  $G$  and is denoted by  $E(G)$ . Moreover, each element of  $E(G)$  is called an *edge* of  $G$  and the order of  $E(G)$  is called the *size* of the graph  $G$ . Furthermore, for any edge  $uv \in E(G)$ , the vertices  $u$  and  $v$  are said to be *adjacent* to each other in  $G$  or *neighbors* of each other in  $G$ ; and the edge  $uv$  is said to be *incident with* the vertices  $u$  and  $v$ . See Figure 2.1e and 2.1f for examples of undirected graphs.

If the arc set of any of the above types of graphs does not contain any loops, then we prefix its name by the word *simple*. For example, see Figures 2.1d and 2.1f for examples of simple oriented and simple undirected graphs, respectively. In this thesis, unless otherwise mentioned, we shall deal only with simple undirected graphs which we shall just call *graphs* from now on.

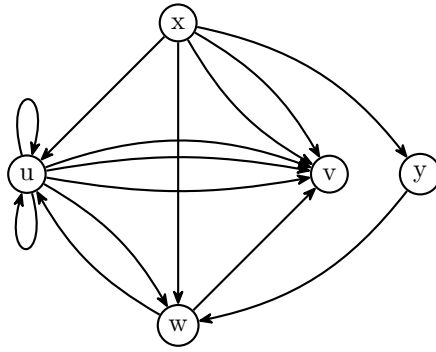
## Graph homomorphisms

Let  $G$  and  $H$  be two graphs. Then a function  $f : V(H) \rightarrow V(G)$  is called a *graph homomorphism*, or simply, a *homomorphism*, from  $H$  to  $G$  if, for any  $u, v \in V(H)$ ,  $f(u)f(v)$  is an edge in  $G$  if  $uv$  is an edge in  $H$ . In symbol, a homomorphism from a graph  $H$  to a graph  $G$  is denoted by  $f : H \rightarrow G$  or, simply by  $H \rightarrow G$ , when the function is either not important or is clear from the context. If a homomorphism  $f : H \rightarrow G$  is such that the function  $f : V(H) \rightarrow V(G)$  is injective (respectively, surjective and bijective), then  $f$  is called an *injective homomorphism* (respectively, a *surjective homomorphism* and a *bijective homomorphism*). If  $f : H \rightarrow G$  is a bijective homomorphism such that the inverse function  $f^{-1} : V(G) \rightarrow V(H)$  is also a homomorphism, then  $f$  is called an *isomorphism* between  $H$  and  $G$ . If there exists an isomorphism  $f$  between two graphs  $H$  and  $G$ , then the graphs are said to be *isomorphic* to each other and, in symbol, is denoted by  $f : H \cong G$ , or simply, by  $H \cong G$ . On the other hand, if no isomorphism exists between two given graphs, the latter are said to be *non-isomorphic* to each other.

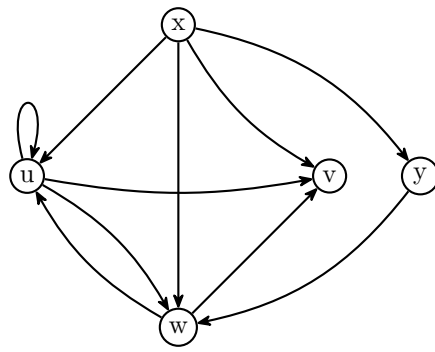
## Subgraphs and distances

Let  $G$  be a graph. A graph  $H$  is called a *subgraph* of  $G$  if  $V(H) \subseteq V(G)$  and  $E(H) \subseteq E(G)$ . If  $H$  is a subgraph of  $G$ , the latter is called a *supergraph* of the former. Moreover, a subgraph  $H$  of  $G$  is called an *induced subgraph* of  $G$  if, for  $u, v \in V(H)$ , the pair  $uv$  is an edge of  $H$  if and only if  $uv$  is an edge of  $G$ . Given a vertex subset  $S$  of  $G$ , we denote by  $G[S]$  the induced subgraph of  $G$  such that  $V(G[S]) = S$ . Moreover,  $G[S]$  is called the subgraph of  $G$  *induced by*  $S$ . For any two graphs  $G$  and  $H$ , if the graph  $G$  does not contain any induced subgraph isomorphic to  $H$ , then  $G$  is called  *$H$ -free*. For any two vertex subsets  $A$  and  $B$  of  $G$ , let  $E[G; A, B] = \{uv \in E(G) : u \in A, v \in B\}$ . Notice that  $E[G; A, B] = E[G; B, A]$ . Also notice that, for any vertex subset  $S$  of  $G$ , the edge set of  $G[S]$  is  $E[G; S, S]$ . Whenever a vertex subset of  $G$  is singleton, say  $\{a\}$ , then for notational convenience, we simply write  $E[G; a, B]$  instead of  $E[G; \{a\}, B]$ .

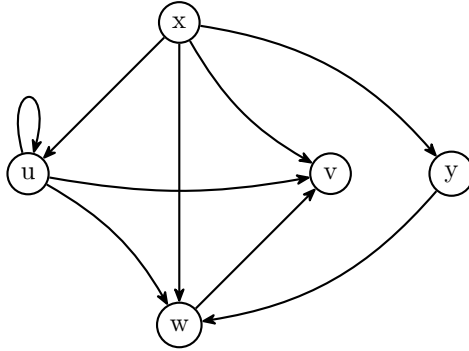
A graph  $G$  on  $n$  vertices such that  $V(G) = \{v_1, v_2, \dots, v_n\}$  and  $E(G) = \{v_i v_{i+1} : i \in [n-1]\}$  is called an  *$n$ -path*, or simply a *path* on  $n$  vertices. An  $n$ -path is usually denoted by  $P_n$ . Each of the vertices  $v_1$  and  $v_n$  is called an *endpoint* of the path  $P_n$  and the integer  $n-1$  is called the *length* of  $P_n$ . Given any two distinct vertices  $u, v$  of a graph  $G$ , let  $d_G(u, v) = \min\{k \in \mathbb{N} : \text{there exists } f : P_k \cong H \text{ from } u \text{ to } v\}$ .



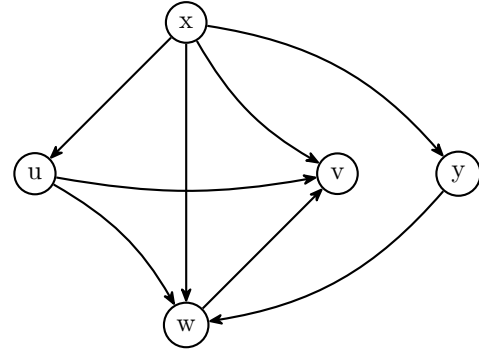
(a) Example of a directed multigraph  $G$  on 5 vertices with vertex set  $V(G) = \{u, v, w, x, y\}$ . Moreover, we have  $A_1(G) = \{uv\}$ ,  $A_2 = \{uv, xv, uu\}$ ,  $A_3 = \{uv, wu, wu, xu, xv, wv, xw, xy, yw, uu\}$ . The arc  $uu$  is a loop, the vertex  $x$  is a source and the vertex  $v$  is a sink.



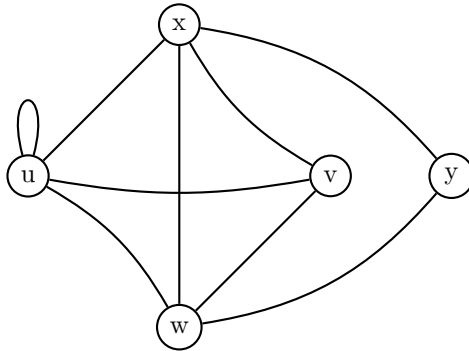
(b) Example of a directed graph  $G$  on 5 vertices with vertex set  $V(G) = \{u, v, w, x, y\}$  and arc set  $A(G) = \{uv, wu, wu, xu, xv, wv, xw, xy, yw, uu\}$ . The arc  $uu$  is a loop, the vertex  $x$  is a source and the vertex  $v$  is a sink.



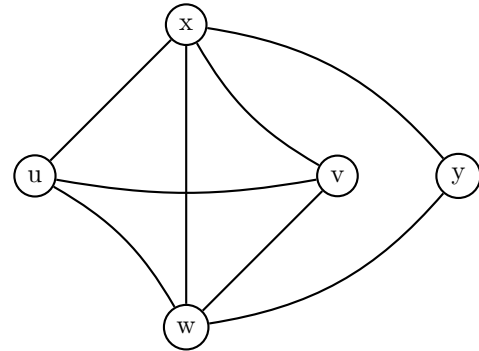
(c) Example of an oriented graph  $G$  on 5 vertices with vertex set  $V(G) = \{u, v, w, x, y\}$  and arc set  $A(G) = \{uv, wu, xu, xv, wv, xw, xy, yw, uu\}$ . The arc  $uu$  is a loop, the vertex  $x$  is a source and the vertex  $v$  is a sink.



(d) Example of a simple oriented graph  $G$  on 5 vertices with vertex set  $V(G) = \{u, v, w, x, y\}$  and arc set  $A(G) = \{uv, wu, xu, xv, wv, xw, xy, yw\}$ . The vertex  $x$  is a source and the vertex  $v$  is a sink.



(e) Example of an undirected graph  $G$  on 5 vertices with vertex set  $V(G) = \{u, v, w, x, y\}$  and edge set  $E(G) = \{uv, wu, xu, xv, wv, xw, xy, yw, uu\}$ . The edge  $uu$  is a loop.



(f) Example of a simple undirected graph  $G$  on 5 vertices with vertex set  $V(G) = \{u, v, w, x, y\}$  and edge set  $E(G) = \{uv, wu, xu, xv, wv, xw, xy, yw\}$ .

Figure 2.1: Examples of several category of graphs.

some induced subgraph  $H$  of  $G$ ; and  $u$  and  $v$  are images of the endpoints of  $P_k$  under  $f$ . Then, the integer  $d_G(u, v)$  is called the *distance* between the vertices  $u$  and  $v$  in  $G$ . It can be checked that  $d_G(u, v) = 1$  if and only if  $uv \in E(G)$ . Moreover, by convention, we let  $d_G(u, v) = 0$  if and only if  $u = v$ . Finally, for any graph  $G$ , we define the parameter  $\text{diam}(G) = \max\{d_G(u, v) : u, v \in V(G)\}$  to be the *diameter* of the graph  $G$ .

## Neighborhoods and domination

Let  $G$  be a graph. For a vertex  $v$  of  $G$ , the set  $N_G[v] = \{u \in V(G) : d_G(u, v) \leq 1\}$  is called the *closed neighborhood* of  $v$  in  $G$ . Moreover, the set  $N_G(v) = N_G[v] \setminus v$  is called the *open neighborhood* (or simply, the *neighborhood*) of  $v$  in  $G$ . We recall that a vertex  $v$  of  $G$  is said to *dominate* all vertices in  $N_G[v]$  and *total-dominate* all vertices in  $N_G(v)$ . Moreover, given a vertex  $v$  and a vertex subset  $S$  of  $G$ , the set  $S$  is said to *dominate* (respectively, *total-dominate*) the vertex  $v$  if there exists a vertex  $u$  in  $S$  that dominates (respectively, total-dominates)  $v$ . Also, for any vertex subset  $S$  of  $G$ , let  $N_G(S) = \bigcup_{s \in S} N_G(s)$  and  $N_G[S] = \bigcup_{s \in S} N_G[s]$ . Then, a vertex subset  $S$  of  $G$  is called a *dominating set* if  $N_G[S] = V(G)$  and is called a *total-dominating set* of  $G$  if  $N_G(S) = V(G)$ . Moreover, the minimum of the cardinalities of all dominating sets of a graph  $G$  is called the *domination number* of  $G$  and is denoted by  $\gamma(G)$ . Similarly, the minimum of the cardinalities of all total-dominating sets of a graph  $G$  is called the *total-domination number* of  $G$  and is denoted by  $\gamma_t(G)$ .

The following observation verifies that these definitions of dominating and total-dominating sets are equivalent to the ones used in Chapter 1.

**Observation 2.1.** *Let  $S$  be a vertex subset of a graph  $G$ . Then, the following assertions are true.*

- (1)  *$S$  is a dominating set of  $G$  if and only if  $N_G[v] \cap S \neq \emptyset$  for all vertices  $v \in V(G)$ .*
- (2)  *$S$  is a total-dominating set of  $G$  if and only if  $N_G(v) \cap S \neq \emptyset$  for all vertices  $v \in V(G)$ .*

See Figure 2.2a and 2.2b for examples of a dominating and a total-dominating set of a graph.

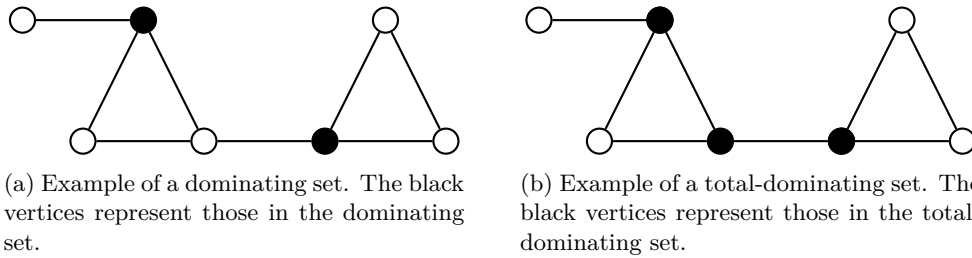


Figure 2.2: Examples of dominating and total-dominating sets of a graph  $G$ . For this graph, we have  $\gamma(G) = 2$  and  $\gamma_t(G) = 3$ .

**Remark 2.2.** *A dominating set of a graph  $G$  always exists.*

**Remark 2.3.** *A total-dominating set of a graph  $G$  exists if and only if  $N_G(v) \neq \emptyset$  for all  $v \in V(G)$ , that is, every vertex of  $G$  has at least one neighbor in  $G$ .*

Let  $v$  be a vertex of a graph  $G$ . If  $N_G[v]$  is a singleton set, that is  $N_G[v] = \{v\}$ , then the vertex  $v$  is called a *domination-forced vertex*. Notice that the only vertices of a graph  $G$  that can be domination-forced are those without any edges incident with them. Similarly, if  $N_G(v)$  is a singleton set, then the vertex in  $N_G(v)$  is called a *total-domination-forced vertex*. Again notice that a vertex  $v$  of a graph  $G$  can be total-domination-forced only if  $v$  is the only neighbor of another vertex of  $G$ . A vertex  $v$  of  $G$  that is either a domination-forced vertex or a total-domination-forced vertex is sometimes simply referred to as a *forced vertex* when the type of domination is clear from the context.

A total-dominating set has also sometimes been called an *open-dominating set* in the literature (see for example, [190, 191]). This is to signify that, by Observation 2.1(2), a total-dominating set is

defined by means of open neighborhoods of vertices. By analogy, as in Observation 2.1(1), since a dominating set is defined by means of closed neighborhoods of vertices, a dominating set can also be called a *closed-dominating set*. In what follows, we shall often have the need to invoke the notion of “open neighborhoods” in the context of total-dominating sets and, similarly, the notion of “closed neighborhoods” in the context of dominating sets. To that end, in order to bring about a unified writing style, we shall often refer to a total-dominating set as an *O-dominating set*; and refer to a dominating set as a *C-dominating set*. We would like to mention here that the letters “O” and “C” will play a significant role in many of the descriptions and notions to follow. They will serve as shorthands to indicate whether a particular concept is defined on the basis of open neighborhoods or closed neighborhoods of vertices. As a result, we let  $\text{NBD-TYPE} = \{C, O\}$  from now on. Therefore, a B-dominating set for an appropriate  $B \in \text{NBD-TYPE}$  will now refer to either of the two types of dominating sets. In addition, a B-dominating set is also said to have the *domination (property of) type B*.

We also introduce some new notations for open and closed neighborhoods of vertices in terms of the shorthands “O” and “C” as  $N_O(G; v) = N_G(v)$  and  $N_C(G; v) = N_G[v]$ , where  $v$  is a vertex of a graph  $G$ . These new notations are handy in the sense that while addressing a general code  $S$ , on several occasions we shall not have the need to mention in detail whether its domination property or its separation property is defined in terms of open neighborhoods or closed neighborhoods of vertices. In such cases, wherever necessary, it is useful to simply mention  $N_B(G; v)$  for some appropriate  $B \in \text{NBD-TYPE}$ . Moreover, we call  $N_B(G; v)$  a *v-B-neighborhood* in  $G$ . Sometimes, when the vertex  $v$  and the  $B \in \text{NBD-TYPE}$  are clear from the context, we may refer to a *v-B-neighborhood* by either a *v-neighborhood* or a *B-neighborhood* or, simply, a *neighborhood*. We also remark here that the usage of the notations  $N_O(G; v)$  and  $N_C(G; v)$  are rather temporary and are restricted to only those parts of this thesis (mostly until Chapter 3) where we address several codes together without always detailing out their differences. In the remaining chapters of the thesis, where we mostly deal with a specific code / separation type in each chapter, we shall revert back to the classical notations  $N_G(v)$  and  $N_G[v]$ , respectively, for these neighborhoods. Thus, combining Observation 2.1 with these newly introduced notations for neighborhoods of vertices, we have the following remark.

**Remark 2.4.** Let  $G$  be a graph and let  $S$  be a vertex subset of  $G$ . Moreover, let  $B \in \text{NBD-TYPE}$ . Then,  $S$  is a *B-dominating set* of  $G$  if and only if  $N_B(G; v) \cap S \neq \emptyset$  for all vertices  $v \in V(G)$ .

## Degrees of vertices

For any vertex  $v$  of a graph  $G$ , let  $\deg_G(v) = |N_G(v)|$  denote the *degree* of  $v$  in  $G$ . Let  $\Delta(G) = \max\{\deg_G(v) : v \in V(G)\}$  be called the *maximum degree* of the graph  $G$ . Similarly, let  $\delta(G) = \min\{\deg_G(v) : v \in V(G)\}$  be called the *minimum degree* of  $G$ . A vertex of degree 0 in  $G$  is called an *isolated vertex* of  $G$ . Moreover, a graph without any isolated vertices is called *isolate-free*. A vertex of degree 1 in  $G$  is called a *leaf* or a *pendant vertex* of  $G$ . Moreover, the only neighbor of a pendant vertex  $v$  of  $G$  is called the *support vertex* of  $v$  in  $G$ .

A graph with maximum degree 3 is called *subcubic*. For a positive integer  $r$ , if a graph  $G$  is such that every vertex of  $G$  is of degree  $r$ , then  $G$  is called an *r-regular graph*. Note that a connected 2-regular graph is a cycle. A 3-regular graph is called a *cubic graph*.

## Other graph terminologies

Let  $G$  be a graph. The graph  $\overline{G}$  with  $V(\overline{G}) = V(G)$  and  $E(\overline{G}) = \{uv : u, v \in V(G), uv \notin E(G)\}$  is called the *complement* of  $G$ . A graph  $G$  is called *complete* if  $E(G) = \{uv : u, v \in V(G), u \neq v\}$ , that is, if any two distinct vertices of  $G$  have an edge between them. A complete graph on  $n$  vertices is usually denoted by  $K_n$ . Moreover, if  $S$  is a vertex subset of a graph  $G$  such that  $G[S]$  is complete, then the set  $S$  is called a *clique*. In particular, for any positive integer  $n$ , the set  $V(K_n)$  is a clique. A graph is called a *diamond* if it is isomorphic to a  $K_4$  minus an edge. A vertex subset  $S$  of a graph

$G$  is called an *independent set* if  $\overline{G[S]}$  is a clique, that is, if no two distinct vertices in  $S$  have an edge between them. Again, in particular,  $\overline{K_n}$  is an independent set.

Let  $S_V$  be a vertex subset of a graph  $G$ , let  $S_E$  be an edge subset of  $G$  and let  $S = S_V \cup S_E$ . Then the graph  $G - S$  is given by  $V(G - S) = V(G) \setminus S_V$  and  $E(G - S) = E(G) \setminus (S_E \cup E[G; S_V, V(G)])$ . Moreover,  $G - S$  is said to be obtained by *deleting  $S$  from  $G$* . Moreover, if  $S$  is a singleton set, say  $S = \{x\}$ , then for notational simplicity, we write  $G - x$  instead of  $G - \{x\}$ . A graph  $G$  is called *connected* if for any two distinct vertices  $u, v \in V(G)$ , there exists an induced path in  $G$  whose endpoints are  $u$  and  $v$ . Two non-empty, vertex-disjoint and connected graphs  $G_1$  and  $G_2$  are called *components* of the graph  $G_1 \oplus G_2 = (V(G_1) \cup V(G_2), E(G_1) \cup E(G_2))$ . The latter graph is called the *disjoint union* of  $G_1$  and  $G_2$  and is also said to be *disconnected*. A graph is said to be *k-vertex-connected*, or simply *k-connected*, if for any vertex subset  $S$  of  $G$  such that  $|S| < k$ , the graph  $G - S$  is connected.

Given a graph  $G$  and a graph property  $\Pi$ , a *maximal subgraph with respect to  $\Pi$*  is a subgraph  $H$  of  $G$  such that, for any vertex  $v \in V(G) \setminus V(H)$ , the induced subgraph  $G[V(H) \cup \{v\}]$  does not satisfy the property  $\Pi$ . On the other hand, a *maximum subgraph with respect to  $\Pi$*  is a subgraph  $H$  of  $G$  such that  $V(H') \subseteq V(H)$  for each subgraph  $H'$  of  $G$  satisfying property  $\Pi$ . Similarly, a *minimal subgraph with respect to  $\Pi$*  is a subgraph  $H$  of  $G$  such that, for any vertex  $v \in V(H)$ , the induced subgraph  $H - v$  does not satisfy the property  $\Pi$ . Moreover, a *minimum subgraph with respect to  $\Pi$*  is a subgraph  $H$  of  $G$  such that  $V(H) \subseteq V(H')$  for each subgraph  $H'$  of  $G$  satisfying property  $\Pi$ .

## 2.1.3 Graph families

In this section, we introduce a list of familiar graph families. Some of these graph families will feature in our later chapters on which we study some specific codes.

### $n$ -Cycles

A graph on  $n$  with a vertex set  $\{v_1, v_2, \dots, v_n\}$  and edge set  $\{v_i v_{i+1} : i \in [n-1]\} \cup \{v_n v_1\}$  is called an *n-cycle*, or simply a *cycle* on  $n$  vertices. A cycle is therefore a 2-regular graph. An *n-cycle* is usually denoted by  $C_n$ . If  $n$  is odd, then  $C_n$  is called an *odd cycle* and if  $n$  is even, then  $C_n$  is called an *even cycle*. Let  $G$  be a graph and let there exist an edge subset  $F \subsetneq E(G)$  such that  $G - F$  is isomorphic to an *n-cycle* for some  $n \geq 4$ . Then each edge of  $G$  in  $F$  is called a *chord*. A  $C_3$ -free graph is also interchangeably called *triangle-free*.

The *girth* of a graph  $G$  is the minimum positive integer  $n$  such that an induced subgraph of  $G$  is isomorphic to an *n-cycle*. For example, triangle-free graphs have girth at least 4.

### Trees

A *tree* is a connected graph that has no subgraph isomorphic to an *n-cycle* for any  $n \geq 3$ . A *rooted tree* is a 2-tuple  $(T, r)$ , where  $T$  is a tree and  $r$  is a vertex of  $T$  labeled as *root*. In a tree  $T$ , for every pair of distinct vertices  $u, v \in V(T)$ , there exists a unique path in  $T$  with  $u$  and  $v$  as its endpoints; or else, two distinct paths between  $u$  and  $v$  would lead to  $G$  having an (induced) subgraph isomorphic to an *n-cycle* for some  $n \geq 3$  and thus, contradicting the definition of a tree. In particular therefore, given a rooted tree  $(T, r)$ , every vertex  $v$  of  $T$  has a unique path with  $r$  and  $v$  as its endpoints. Moreover, given a vertex  $u$  of the rooted tree  $(T, r)$ , we also have the following widely used definitions.

- (1) A *child* of  $u$  is a vertex  $v$  of  $T$  such that  $uv \in E(T)$  and  $d_T(v, r) = d_T(u, r) + 1$ . If  $v$  is a child of  $u$ , then, conversely,  $u$  is called a *parent* of  $v$ .
- (2) A *grandchild* of  $u$  is a vertex  $w$  of  $T$  such that there exists another vertex  $v \in V(T)$  with  $uv, vw \in E(T)$  and  $d_T(w, r) = d_T(u, r) + 2$ . If  $w$  is a grandchild of  $u$ , then, conversely,  $u$  is called a *grandparent* of  $w$ .



- (3) A *great-grandchild* of  $u$  is a vertex  $x$  of  $T$  such that there exist vertices  $v, w \in E(T)$  with  $uv, vw, wx \in E(T)$  and  $d_T(x, r) = d_T(u, r) + 3$ . If  $x$  is a great-grandchild of  $u$ , then, conversely,  $u$  is called a *great-grandparent* of  $x$ .

See Figure 2.3 for an example of a tree. A graph whose every component is a tree is called a *forest*. Quite often, a vertex of a tree is also interchangeably called a *node*.

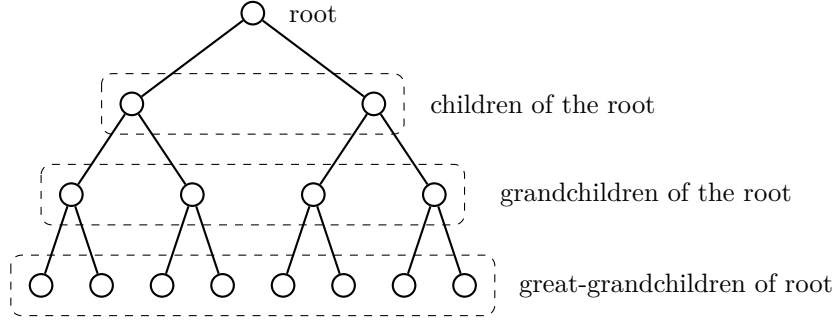


Figure 2.3: Example of a tree showing the children, grandchildren and the great-grandchildren of the root of the tree.

## Bipartite graphs

A graph  $G$  is called *bipartite* if its vertex set can be partitioned into two parts, say  $U$  and  $V$ , and  $E(G) = E[G; U, V]$ . In other words, in a bipartite graph  $G$ , we have  $E[G; U, U] = E[G; V, V] = \emptyset$ , that is,  $G$  has no edges between pairs of vertices both belonging to either  $U$  or  $V$ . Figure 7.2a is an example of a bipartite graph. A bipartite graph has no odd cycle as a subgraph, or else, it would lead to an edge between two vertices of the same part, thus contradicting the definition. Trees are examples of bipartite graphs. A bipartite graph on  $n + 1$  vertices is called an *n-star* (or simply a *star*) if one of the parts of its vertex set is of cardinality 1. An  $n$ -star is usually denoted by  $K_{1,n}$ . A  $k_{1,3}$  is also called a *claw*. Notice that a star is also a tree.

## Half-graphs

Half-graphs are a subfamily of bipartite graphs which are defined as follows. For any integer  $k \geq 1$ , the *half-graph*  $B_k = (U \cup W, E)$  is the bipartite graph with vertices in  $U = \{u_1, \dots, u_k\}$ ,  $W = \{w_1, \dots, w_k\}$  and edges  $u_i w_j$  if and only if  $i \leq j$ . See Figure 7.2a for example of the half-graph  $B_4$ . In particular, we have  $B_1 \cong K_2$  and  $B_2 \cong P_4$ . Half-graphs were studied and first named so by Erdős and Hanjal (see for example, [84]).



(a)  $B_4$ : Half-graph (also an example of a bipartite graph).

(b)  $\overline{B_4}$ : Complement of half-graph (also an example of a cobipartite graph).

Figure 2.4: Examples of bipartite and cobipartite graphs

## Cobipartite graphs

A graph is called *cobipartite* if its complement is a bipartite graph. In other words, a cobipartite graph is one whose vertex set can be partitioned into two parts, each of which is a clique. See Figure 2.4b for an example of a cobipartite graph, namely, the complement of the half-graph  $B_4$ .

## Chordal graphs

A graph  $G$  is called *chordal* if every subgraph  $H$  of  $G$  isomorphic to an  $n$ -cycle with  $n \geq 4$  has a chord. In other words, a chordal graph  $G$  has no induced subgraph isomorphic to an  $n$ -cycle with  $n \geq 4$ . Trees, complete graphs, independent sets are all examples of chordal graphs. Figure 2.5 is also an example of a chordal graph; and so are the graphs in Figures 2.6a and 2.6b.

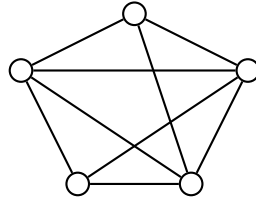


Figure 2.5: Example of a chordal graph.

## Split graphs

Split graphs form a subfamily of chordal graphs. A *split graph*  $G$  is one whose vertex set can be partitioned into two parts, say  $U$  and  $V$ , such that  $U$  is an independent set and  $G[V] \cong K_{|V|}$ . Figure 2.6a and 2.6b are examples of split graphs.

## Spiders

A *headless spider* is a split graph (and hence, also a chordal graph) with  $V = \{q_1, \dots, q_k\}$  which is a clique and  $U = \{s_1, \dots, s_k\}$  which is an independent set. In addition, a headless spider is *thin* (respectively, *thick*) if  $s_i$  is adjacent to  $q_j$  if and only if  $i = j$  (respectively,  $i \neq j$ ). By definition, it is clear that the complement of a thin headless spider  $H_k$  is a thick headless spider  $\overline{H}_k$ , and vice-versa. We also have  $H_2 \cong \overline{H}_2 \cong P_4$  and the two headless spiders  $H_3$ , also called a *net*, and  $\overline{H}_3$ , also called a *sun*, are depicted in Figures 2.6a and 2.6b, respectively.

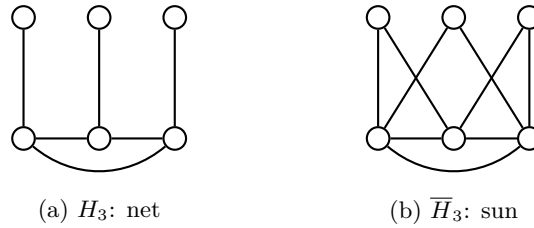


Figure 2.6: Examples of thin and thick headless spiders.

## Block graphs

Block graphs form a subfamily of chordal graphs. A *block graph* is a graph in which every maximal 2-connected subgraph is complete. In a block graph, every maximal complete subgraph is called a *block*.

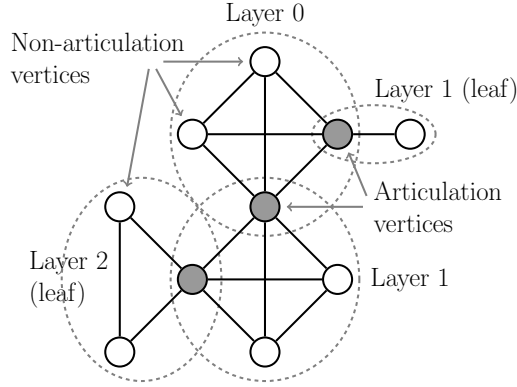


Figure 2.7: Example of different layer numbers, articulation vertices (gray) and non-articulation vertices (white) of a connected block graph.

**Remark 2.5** ([17]). *A graph is a block graph if and only if it is a diamond-free chordal graph.*

**Remark 2.6.** *The vertex sets of any two distinct blocks of a block graph  $G$  intersect in at most one vertex of  $G$ .*

We next establish here some notations and terminologies specific to block graphs.

**Notations and terminologies concerning block graphs.** For a block graph  $G$ , we let  $\mathcal{K}(G)$  denote the set of all blocks of  $G$ . Noting by Remark 2.6 that the vertex sets of any two distinct blocks  $K$  and  $K'$  of  $G$  intersect in at most a single vertex, any vertex  $x \in V(G)$  such that  $\{x\} = V(K) \cap V(K')$  is called an *articulation vertex* of both  $K$  and  $K'$ . We define  $\text{art}(K)$  to be the set of all articulation vertices of a block  $K \in \mathcal{K}(G)$ . For a connected block graph, we fix a *root block*  $K_0 \in \mathcal{K}(G)$  and define a system of assigning numbers to every block of  $G$  depending on “how far” the latter is from  $K_0$ . So, define a *layer function*  $f : \mathcal{K}(G) \rightarrow \mathbb{Z}$  on  $G$  by:  $f(K_0) = 0$ , and for any other  $K (\neq K_0) \in \mathcal{K}(G)$  (also called a *non-root block*), define inductively  $f(K) = i$  if  $V(K) \cap V(K') \neq \emptyset$  for some block  $K' (\neq K) \in \mathcal{K}(G)$  such that  $f(K') = i - 1$ . For a pair of blocks  $K, K' \in \mathcal{K}(G)$  such that  $f(K) = f(K') + 1$ , define  $\text{art}^-(K) = V(K) \cap V(K')$ ; and for the root block  $K_0$ , define  $\text{art}^-(K_0) = \emptyset$ . Note that for a block  $K \in \mathcal{K}(G)$  such that  $f(K) \geq 1$ , we have  $|\text{art}^-(K)| = 1$ , and the only vertex in  $\text{art}^-(K)$  is called the *negative articulation vertex* of the block  $K$ . In contrast to the negative articulation vertices of  $G$ , define  $\text{art}^+(K) = \text{art}(K) \setminus \text{art}^-(K)$  to be the set of all *positive articulation vertices* of the block  $K$  and  $\overline{\text{art}}(K) = V(K) \setminus \text{art}(K)$  to be the set of all *non-articulation vertices* of  $K$ . Any block  $K$  with  $|\text{art}(K)| = 1$  is called a *leaf block* and all blocks that are not leaf blocks are called *non-leaf blocks*. Note that a block graph always has leaf blocks. For simplicity, we also denote the set  $f^{-1}(\{i\})$  by  $f^{-1}(i)$ . Then, for each  $i \geq 0$ ,  $f^{-1}(i)$  is called the  *$i$ -th layer* of  $G$  and each block  $K \in f^{-1}(i)$  is said to be *in the  $i$ -th layer* of  $G$ . See Figure 2.7 for an illustration of the layers and the related concepts in a connected block graph.

## Interval graphs

An *interval graph*  $G$  is a graph that can be defined in the following manner. Let  $V(G) = \{v_1, v_2, \dots, v_n\}$ . Then, for each  $j \in [n]$ , there exists an interval  $I_j$  of  $\mathbb{R}$  such that  $E(G) = \{v_j v_k : j, k \in [n], j \neq k \text{ and } I_j \cap I_k \neq \emptyset\}$ . An interval graph is called a *unit interval graph* if  $|I_j| = 1$  for each  $j \in [n]$ . See Figures 2.8a and 2.8b for examples of an interval and a unit interval graph, respectively.

## Permutation graphs

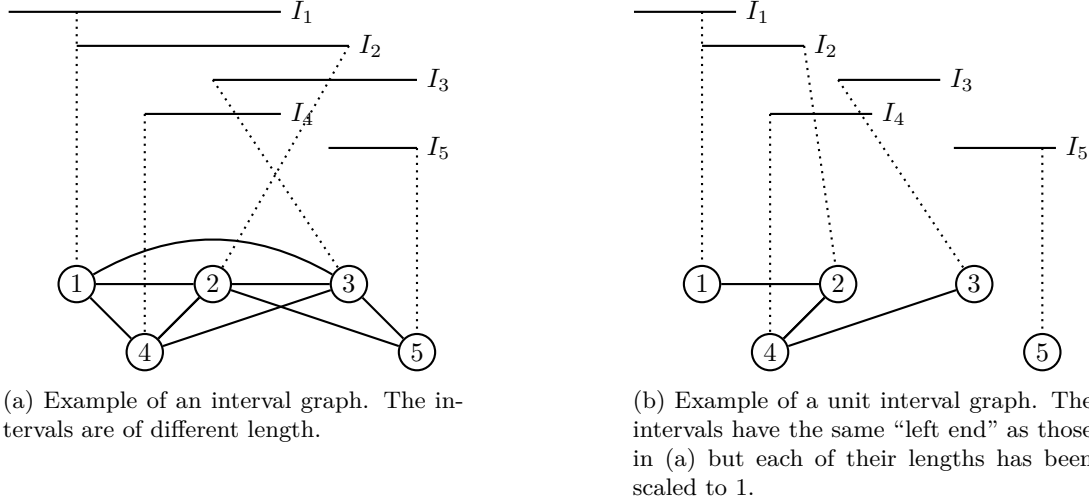


Figure 2.8: Example of interval and unit interval graphs. The vertex  $j$  of the graphs constructed corresponds to the interval  $I_j$  (this correspondence has been indicated by the dotted lines).

For any positive integer  $n$ , a *permutation* on the set  $[n]$  is a bijective function  $\sigma : [n] \rightarrow [n]$ . For any  $i \in [n]$ , let  $\sigma_i = \sigma(i)$ . A *permutation graph*  $G$  is a graph that can be defined in the following manner. Let  $V(G) = \{v_1, v_2, \dots, v_n\}$ . Then, there exists a permutation  $(\sigma_1 \sigma_2 \dots \sigma_n)$  on  $[n]$  such that  $E(G) = \{v_i v_j : i < j \text{ and } \sigma_i < \sigma_j, \text{ where } \sigma_i = j \text{ and } \sigma_j = i\}$ . See Figure 2.9 for the example of a permutation graph on 5 vertices.

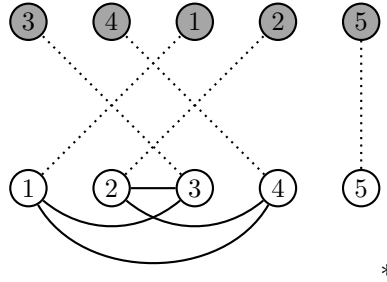


Figure 2.9: Example of a permutation graph on 5 vertices. The permutation  $\sigma : [5] \rightarrow [5]$  is given by  $\sigma(j) = \text{number on the shaded vertex straight above the white vertex } j$ .

## Line graphs

Given a graph  $G$ , the *line graph*  $L(G)$  of  $G$  is given by  $V(L(G)) = \{v(e) : e \in E(G)\}$  and  $E(L(G)) = \{v(e)v(e') : e = uv \in E(G), e' = u'w' \in E(G) \text{ and } \{u, w\} \cap \{u', w'\} \neq \emptyset\}$ . See Figure 2.10 for example of the line graph of the 3-star.

## Suns

A *sun* is a graph  $G = (C \cup S, E)$  whose vertex set can be partitioned into  $S$  and  $C$ , where, for an integer  $k \geq 3$ , the set  $S = \{s_1, \dots, s_k\}$  is an independent set and  $C = \{c_1, \dots, c_k\}$  is a (not necessarily chordless) cycle. A *thin sun*  $T_k = (C \cup S, E)$  is a sun where  $s_i$  is adjacent to  $c_j$  if and only if  $i = j$ . Therefore, thin headless spiders are special thin suns where all chords of the cycle  $C$  are present (such that  $C$  is a clique). Other special cases of thin suns are *sunlets* where no chords of the cycle  $C$  are present (such that  $C$  induces a cycle). For illustration, for  $k = 3$ , the (only) thin

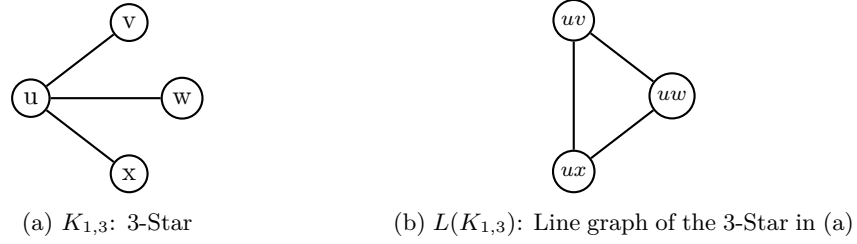
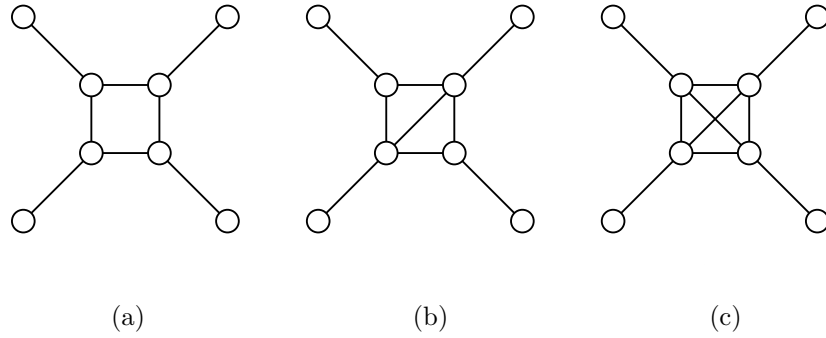


Figure 2.10: Example of line graph of a 3-star.

sun  $T_3$  is the thin headless spider  $H_3$  (see Figure 2.6a); for  $k = 4$ , the three possible thin suns  $T_4$  are depicted in Figure 2.11.

Figure 2.11: The three thin suns  $T_4$ , where (a) is a sunlet and (c) a thin headless spider.

## Planar and outerplanar graphs

A *planar graph* is one which is possible to draw on a plane without any of its edges intersecting. Such a drawing of a planar graph is called a (*planar*) *embedding* of  $G$  in the plane. For example, all three graphs in 2.11 are planar graphs. Given a planar graph  $G$  with an embedding in the plane, a *face* of  $G$  is a region of the plane that is either bounded (area-wise) by the edges of  $G$  or is the *unique* unbounded region of the plane which is the complement (set-wise) of the area occupied by  $G$  in the plane. In an embedding of a planar graph  $G$  in the plane, a vertex of  $G$  may either be on a bounded face or on the unbounded face. An *outerplanar graph*  $G$  is a planar graph with an embedding in the plane such that every vertex of  $G$  is on the unbounded face. For example, the first two graphs in Figure 2.11 are outerplanar graphs.

## Series-parallel graphs

An  $(s, t)$ -*series-parallel graph* is any graph (with two distinguished vertices  $s$  and  $t$ ) recursively defined as follows. An edge  $st$  is an  $(s, t)$ -series-parallel graph. Moreover, given an  $(s_1, t_1)$ -series-parallel graph  $G_1$  and an  $(s_2, t_2)$ -series-parallel graph  $G_2$ , an  $(s, t)$ -series-parallel graph  $G$  can be obtained from the disjoint union of  $G_1$  and  $G_2$  by identifying either:

- (a) *series composition*:  $t_1$  and  $s_2$  (in which case  $s = s_1$ , and  $t = t_2$ ) or,
- (b) *parallel composition*:  $s_1$  and  $s_2$  on the one hand, and  $t_1$  and  $t_2$  on the other hand (in which case  $s = s_1 = s_2$ , and  $t = t_1 = t_2$ ).

## 2.1.4 Computational complexity

A *computational problem*, or simply a *problem*, can be thought of an infinite collection of strings of *inputs* and a certain (fixed) task that is to be carried out on a given input string. An input string of a problem is usually fed to a computing device (like a computer) and, on carrying out the task on such a string, the final outcome is called the *output* to the problem. This process of producing an output is called *solving* the problem. Each time a problem is to be solved, the string of inputs can be varied; and each such distinct input string is called a *problem instance*, or simply an *instance*. In other words, varying the input string each time a problem is to be solved, the latter can have an infinite number of instances. On the other hand, the set of tasks to be carried out on each instance is fixed for the problem.

Therefore, more formally, a problem  $P$  is defined as a 2-tuple  $(\mathcal{I}, \mathcal{Q})$ . The set  $\mathcal{I}$  is collection of *instances* and  $\mathcal{Q}$  is a *task* (or a *question*) associated with the problem  $P$  and which acts on an instance to produce an *output*. To carry out the task  $\mathcal{Q}$ , one usually devises an *algorithm* which is a set of even smaller tasks, or instructions, that are to be carried out in a step-by-step manner to arrive at the output. We refer the reader to consult the book [71] by Cormen et al. for all standard terminologies in this topic.

The domain of *computational complexity* is aimed at segregating problems according to their “level of difficulty” in solving them. In this context, the *complexity* of an algorithm is the amount of resources — for example, in time or memory — that the algorithm uses in order to attain the outcome to the problem. With that, the *complexity* of a problem  $P$  is defined as the complexity of the “best possible” algorithm, that is, one that solves  $P$  using the least amount of resources. The more widely studied complexity type is that of *time-complexity* of an algorithm or a problem whereby the resource type is taken to be time. Therefore, there is a common consensus within the scientific community to equate this “level of difficulty” of a problem to the time-complexity of the problem.

Given a problem  $P = (\mathcal{I}, \mathcal{Q})$ , let each instance of the problem be denoted by the letter  $I$ . In other words,  $\mathcal{I} = \{I : I \text{ is a problem instance of } P\}$ . The running time of an algorithm  $\mathcal{A}$  to solve  $P$  is usually determined in terms of  $|I|$ , where the latter is referred to as the *input size* of the problem. Intuitively speaking, the input size  $|I|$  to a problem is the amount of information — usually specified in terms of the number of 0/1-bits — required to meaningfully describe the input to the medium, for example a computer, that aims to solve it. We should also point out here that by running time of an algorithm, we mean its *worst-case* running time. This is because, very often, the set of instructions laid out by an algorithm to solve a problem define a method to explore several options — either by means of loops or if-else decisions — available within the scope of the problem instance. Hence, in practice, an algorithm may have to exhaust all such possible options before arriving at the output and thus, attaining its worst-case running time.

Cobham in [66] and Edmonds in [83] had recommended the idea of evaluating the efficiency of an algorithm  $\mathcal{A}$  by whether or not the worst-case running time of  $\mathcal{A}$  is polynomial in  $|I|$ . If the running time of  $\mathcal{A}$  is polynomial in  $|I|$ , then the algorithm is, generally speaking, regarded as “efficient”; and otherwise, not.

The kind of computational problems that we shall deal with in this thesis are called *decision problems* whereby the output to the problem has exactly *two* possible values: either a “YES” or a “NO”. If the output to a problem  $P$  corresponding to an instance  $I$  is YES, then  $I$  is called a *YES-instance* of  $P$ ; or else,  $I$  is called a *NO-instance* of  $P$ . We now provide some examples of decision problems that we shall also use later in the thesis.

One of the most important problems in the context of computational complexity is that of SAT — or the *Boolean satisfiability problem*. In SAT, an instance  $I$  is a 2-tuple  $(X, \mathcal{C})$ .  $X$  is a set of *boolean variables*, that is, variables which assume values only in 0 and 1. If  $x \in X$ , then  $\neg x$  is the variable such that  $(x + \neg x) \bmod 2 \equiv 1$ . Any element of the set  $X \cup \{\neg x : x \in X\}$  is called a *literal* (from the variable set  $X$ ). On the other hand,  $\mathcal{C}$  is a set of *clauses* such that each clause is a formula in

terms of some literals from  $X$  and the logical operator OR ( $\vee$ ) defined by  $a \vee b = (a + b) \bmod 2$  for any  $a, b \in \{0, 1\}$ . An example of such a clause would be  $\mathbf{c} = x_1 \vee x_2 \vee \neg x_3 \vee \neg x_4 \vee x_5$ , where  $x_i \in X$  for  $i \in [5]$ .

Let  $|X| = n$  and  $|\mathcal{C}| = m$ . Moreover, let  $X = \{x_1, x_2, \dots, x_n\}$  and  $\mathcal{C} = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_m\}$ . The logical operator AND ( $\wedge$ ) is defined to be the function  $\wedge : \{0, 1\} \rightarrow \{0, 1\}$  given by  $a \wedge b = (a \times b) \bmod 2$  for any  $a, b \in \{0, 1\}$ . Given an instance  $(X, \mathcal{C})$ , a *boolean formula* with respect to  $X$  and  $\mathcal{C}$  is defined by  $\phi = \mathbf{c}_1 \wedge \mathbf{c}_2 \wedge \dots \wedge \mathbf{c}_m$ . An *assignment* on  $X$  is an allotment of 0/1-values to the variables in  $X$ ; that is, an assignment is an  $n$ -tuple from  $\{0, 1\}^n$ . Moreover, any  $n$ -tuple for which the value of the formula  $\phi$  is 1, is said to *satisfy*  $\phi$  or to be a *satisfying assignment* on  $X$ . It can be verified that an assignment on  $X$  is a satisfying one if and only if each clause in  $\mathcal{C}$  contains at least one literal that has been allotted the value 1 by the assignment.

Given an instance  $(X, \mathcal{C})$  to SAT, the question related to the problem is whether or not there exists a satisfying assignment on  $X$ . We now formally define SAT as a decision problem.

**SAT**

**Input:** A set  $X$  of boolean variables and a set  $\mathcal{C}$  of clauses defined in terms of the literals from  $X$ .

**Question:** Does there exist a satisfying assignment on  $X$ ?

One of the most well-known versions of SAT is called 3-SAT and is defined as follows.

**3-SAT**

**Input:** A set  $X$  of boolean variables and a set  $\mathcal{C}$  of clauses defined in terms of the literals from  $X$  such that each clause contains at most 3 literals.

**Question:** Does there exist a satisfying assignment on  $X$ ?

Replacing 3 by any positive integer  $k$  in 3-SAT, we get the more general version of the problem called  $k$ -SAT.

**$k$ -SAT**

**Input:** A set  $X$  of boolean variables and a set  $\mathcal{C}$  of clauses defined in terms of the literals from  $X$  such that each clause contains at most  $k$  literals.

**Question:** Does there exist a satisfying assignment on  $X$ ?

A further derivative of the 3-SAT is (3,3)-SAT formulated in [197] and is defined as follows.

**(3,3)-SAT**

**Input:** A set  $X$  of boolean variables and a set  $\mathcal{C}$  of clauses defined in terms of the literals from  $X$  such that each clause contains at most 3 literals and each variable appears in at most 3 clauses.

**Question:** Does there exist a satisfying assignment on  $X$ ?

We remark here that if each clause contains *exactly* 3 variables, and each variable appears exactly 3 times, then the problem is polynomial-time solvable [197]. Another restriction of (3,3)-SAT is LINEAR SAT or LSAT and is defined as follows.

**LINEAR SAT (LSAT)**

**Input:** A set  $X$  of boolean variables and a set  $\mathcal{C}$  of clauses defined in terms of the literals from  $X$  such that each clause contains at most 3 literals; each literal from  $X$  can appear in at most two clauses; and any two distinct clauses can contain at most one literal in common.

**Question:** Does there exist a satisfying assignment on  $X$ ?

We now turn our attention to the computational complexity of our central problem in this thesis, that is, given an  $X \in \text{CODES} = \{\text{LD, LTD, ID, ITD, OD, OTD, FD, FTD}\}$  and an  $X$ -admissible graph  $G$ , what is the  $X$ -number of  $G$ ? One way to answer that question is by actually finding a minimum  $X$ -code  $S$  of  $G$  and then we have  $\gamma^X(G) = |S|$ . This would be an *optimization problem* since we are required to minimize the solution size (that is, the order of the code that we find) as much as possible. However, as is very often the norm in studying complexities of optimization problems, we can turn it into a decision problem by simply introducing a parameter, say  $k$ , to the input to the problem. We demonstrate this next by defining the decision problem related to finding a minimum  $X$ -code of an  $X$ -admissible graph.

**X-CODE**

**Input:** An  $X$ -admissible graph  $G$  and an integer  $k$ .

**Question:** Does there exist an  $X$ -code  $S$  of  $G$  such that  $|S| \leq k$ ?

The validity of the input graph  $G$  to X-CODE can be verified in polynomial-time, as it is possible to check if a graph has twins or isolated vertices in time polynomial in the order of  $G$ . Since the vertex set of an  $X$ -admissible graph  $G$  on  $n$  vertices is an  $X$ -code of  $G$ , it follows that X-CODE can be answered with YES in polynomial-time (in  $n$ ) if  $k \geq n$ . Thus, it is reasonable to assume from now on that  $k < n$ . Now, if there exists an algorithm, say  $\mathcal{A}_X$ , which solves X-CODE in polynomial-time, say  $p(n)$ , then by calling this algorithm  $\mathcal{A}_X$  as a subroutine for all  $k \in [n]$ , that is, at most  $n$  times, we can find the minimum value of  $k$  for which  $\mathcal{A}_X$  answers YES. This minimum value of  $k$  would be the  $X$ -number of  $G$  and we would have found it in a running time at most  $n \cdot p(n)$ , which is also polynomial in  $n$ . This demonstrates that decision problems may also be used to arrive at the  $X$ -number of a graph (or, in general, at the optimal value of the corresponding optimization problem) in polynomial-time, given that an algorithm to the decision problem with a polynomial running time already exists. This exhibits the motivation to study decision problems also in the context of optimization problems.

The two *complexity classes* of decision problems that we shall look at in this thesis are those of P and NP. The class P contains all decision problems which can be solved in polynomial-time in the input size of the problem. On the other hand, the class NP contains all decision problems, say  $P$ , such that for an instance  $I$  of  $P$ , there exists what is called a *certificate*  $C(I)$  with  $C(I) \in |I|^{\mathcal{O}(1)}$  and a polynomial-time *verifying algorithm* taking  $(I, C(I))$  as an input and with an output YES if and only if  $I$  is a YES-instance of  $P$ .

It is widely believed that the problems in NP are “harder” than the problems in P. To that end, to establish a correspondence between problems with similar “level of difficulty”, one uses techniques known as *polynomial-time reductions* from one problem to another. The most widely used reductions are those postulated by Karp [148]. In general, a reduction  $\mathcal{R}$  from a problem  $P$  to another problem  $Q$  is an algorithm such that, given an instance  $I$  of  $P$ , it outputs an instance  $J$  of  $Q$  such that the following are true.

- (1)  $|J| \leq |I|^{\mathcal{O}(1)}$ .
- (2)  $I$  is a YES-instance of  $P$  if and only if  $J$  is a YES-instance of  $Q$ .

Moreover, a reduction  $\mathcal{R}$  from  $P$  to  $Q$  is *polynomial-time* if the running time of  $\mathcal{R}$  is polynomial in the input size of  $P$ . With such a polynomial-time reduction, the former problem is viewed to be “not harder than” the latter — here, “hard” refers to not being polynomial-time solvable. This is because, if  $Q$  is “easy”, it has an algorithm, say  $\mathcal{A}_Q$ , which runs in polynomial-time in the input size of  $Q$ . This further implies that one can solve  $P$  by reducing an instance  $I$  of  $P$  to an instance  $J$  of  $Q$  in time  $|I|^{\mathcal{O}(1)}$  and then using the algorithm  $\mathcal{A}_Q$  as a subroutine to solve  $P$  in time

$$|I|^{\mathcal{O}(1)} + |J|^{\mathcal{O}(1)} \leq |I|^{\mathcal{O}(1)} + (|I|^{\mathcal{O}(1)})^{\mathcal{O}(1)} = |I|^{\mathcal{O}(1)} + |I|^{\mathcal{O}(1)} = |I|^{\mathcal{O}(1)}$$

which is polynomial in  $|I|$ . Therefore, this makes  $P$  “not harder than”  $Q$  in terms of being polynomial-time solvable. On the other hand, if the problem  $P$  is already known to be “hard”, then



it makes  $Q$  a “hard” problem as well (or else, by our previous discussion,  $P$  would be polynomial-time solvable thus leading to a contradiction).

We now come to a more rigorous definition of what we mean by a “hard” problem. A problem  $P$  is called *NP-hard* if every problem in NP can be (polynomial-time) reduced to  $P$ . In other words, an NP-hard problem is “as hard as” every other problem in the class NP. Moreover, if an NP-hard problem itself belongs to the class NP, then the problem is said to be *NP-complete*. Thus, NP-complete problems are generally regarded as the “most difficult” problems in the class NP.

SAT was the first problem to be proven as NP-complete by Cook [69] and Levin [158]. Since then, several other problems have been proven to be NP-complete (see the book [109] by Garey and Johnson and [148] by Karp for some selected problems). Among such problems proven to be NP-complete are also various versions of SAT. The above-mentioned problems of  $k$ -SAT for  $k \geq 3$  are all known to be NP-complete (this is implied from the proof by Karp in [148] that 3-SAT is NP-complete). So are the problems of (3,3)-SAT [197] and LSAT [12]. In fact, there is a stronger conjecture concerning 3-SAT (or  $k$ -SAT in general), namely, the *Exponential Time Hypothesis* [139], or ETH for short. The conjecture roughly states that the  $n$ -variable 3-SAT cannot be solved in time  $2^{o(n)}n^{O(1)}$ . More formally, the statement of the hypothesis is the following.

**Conjecture 2.1** (Exponential-Time Hypothesis (ETH) [139]). *For each positive integer  $k$ , let  $s_k = \inf\{\epsilon \in \mathbb{R} : \text{there exists a } O(2^{\epsilon n}) \text{ algorithm for solving } k\text{-SAT}\}$ . Then, for  $k \geq 3$ , we have  $s_k > 0$ .*

### 2.1.5 Parameterized complexity

To cope with the “difficulty” of solving an NP-hard problem in decent running-times, such problems have been studied through the lens of *parameterized complexity*. In this domain, given a problem  $P = (\mathcal{I}, \mathcal{Q})$ , we consider the *parameterized problem* defined by  $\text{para}(P) = (\mathcal{I} \times \mathbb{N}^*, \mathcal{Q})$ . In other words, any instance of  $\text{para}(P)$  is of the form  $(I, \ell)$ , where  $I \in \mathcal{I}$  is an instance of the (original) problem  $P$  and  $\ell$  is any positive integer. The input  $\ell$  to  $\text{para}(P)$  is called a *parameter input*, or simply a *parameter*, to  $\text{para}(P)$ .

If  $\ell$  originates from the formulation of the problem  $P$  then  $\ell$  is usually referred to as a *natural parameter*. On the other hand, if  $\ell$  defines a restriction on all valid instances of  $P$  to a specific subset, then  $\ell$  is often referred to as a *structural parameter*. For example, in X-CODE defined above, the input  $k$  to the problem can be considered as a natural parameter, as it originates from the definition itself. Similarly, the order of the input graph  $G$  may also be considered as a natural parameter as it forms an integral part of the input to the problem. On the other hand, consider the following restriction to an instance of X-CODE: no vertex subset of an input graph to X-CODE can be a clique of cardinality greater than  $\ell$ . In such a case, we restrict all input graphs to X-CODE to have their clique number at most  $\ell$ . In such a case, the parameter  $\ell$  may be considered as a *structural parameter*. We shall define some more structural parameters later in this section.

Let  $I$  be a generic instance to a problem  $P$  and let  $\ell$  be the parameter input to  $\text{para}(P)$ . If the latter problem runs in a time  $f(\ell) \cdot |I|^{O(1)}$ , where  $f$  is some computable function, then the problem  $P$  is said to be *fixed parameter tractable*, or FPT for short. Moreover, the running time of the form  $f(\ell) \cdot |I|^{O(1)}$  is called an *FPT time* and an algorithm for  $\text{para}(P)$  solving the problem in the said time is said to be an *FPT algorithm*. In particular, if  $P$  is a decision problem (which implies that so is  $\text{para}(P)$ ), then an FPT algorithm decides the YES-instance of  $\text{para}(P)$  in FPT time.

As an analogy to classical complexity theory, FPT problems can be thought of as being analogous to problems in P. Intuitively, if a problem  $P$  is FPT, up to a factor of  $f(|I|)$  which may not be polynomial, the (FPT) algorithm of  $\text{para}(P)$  solves  $P$  in a running-time polynomial in the input size  $|I|$ . Similarly, analogous to the class NP which is commonly assumed to be a larger class of problems than P, in the theory of parameterized complexity, Downey and Fellows defined in [81] a sequence of classes, namely, those of  $W[0] = \text{FPT}$ ,  $W[1]$ ,  $W[2]$ ,  $W[3]$  etc., where it is popularly believed that  $W[i] \subsetneq W[j]$  if  $i < j$ . This is commonly referred to as the *W-hierarchy*. It turns out that most

computational problems occur within the first three levels of the  $W$ -hierarchy, that is, within  $W[0]$ ,  $W[1]$  and  $W[2]$ . Thus, the most common way to show that it is unlikely that a parameterized problem admits an FPT algorithm is by showing that the problem is either  $W[1]$ -hard or  $W[2]$ -hard.

As in classical complexity, in the paradigm of parameterized complexity as well, to prove that a problem  $\text{para}(Q)$  is  $W[i]$ -hard, we first take any  $W[i]$ -hard problem, say  $\text{para}(P)$ , and apply a *parameterized reduction* from  $\text{para}(P)$  to  $\text{para}(Q)$ . The said reduction  $\mathcal{R}$  is an algorithm such that, given an instance  $(I, k)$  of  $\text{para}(P)$ , it outputs an instance  $(J, \ell)$  of  $\text{para}(Q)$  with the following being true.

- (1)  $|J| \leq G(k)|I|^{\mathcal{O}(1)}$  for some computable function  $G$ .
- (2)  $\ell \leq g(k)$  for some computable function  $g$ .
- (3)  $I$  is a YES-instance of  $\text{para}(P)$  if and only if  $J$  is a YES-instance of  $\text{para}(Q)$ .
- (4) The running-time of  $\mathcal{R}$  is  $f(k) \cdot |I|^{\mathcal{O}(1)}$  for some computable function  $f$ .

With such a parameterized reduction, if  $Q$  is FPT, it implies that so is  $P$ . This is because, there exists an FPT algorithm, say  $\mathcal{A}_Q$ , which solves  $\text{para}(Q)$  in time  $h(\ell) \cdot |J|^{\mathcal{O}(1)}$ . Thus,  $\text{para}(P)$  can be solved by using the reduction in time  $f(k) \cdot |I|^{\mathcal{O}(1)}$  and then invoking  $\mathcal{A}_Q$  as a subroutine and thus attaining the output in time

$$f(k) \cdot |I|^{\mathcal{O}(1)} + h(\ell) \cdot |J|^{\mathcal{O}(1)} \leq f(k) \cdot |I|^{\mathcal{O}(1)} + h(g(k)) \cdot \left(G(k)|I|^{\mathcal{O}(1)}\right)^{\mathcal{O}(1)} = F(k) \cdot |I|^{\mathcal{O}(1)},$$

where  $F$  is some computable function. In other words, as was in the case of classical complexity,  $\text{para}(Q)$  is “as hard as”  $\text{para}(P)$ .

One of the crucial steps to design parameterized algorithms is to develop some preprocessing / reduction rules that run in polynomial-time to create an equivalent instance. A parameterized problem  $\text{para}(P)$  is said to admit a *kernelization* if, given an instance  $(I, k)$  of  $\text{para}(P)$ , there is an algorithm that runs in time polynomial in  $|I| + k$  and constructs an instance  $(J, \ell)$  of  $\text{para}(P)$  such that

- (1)  $(I, k) \in \text{para}(P)$  if and only if  $(J, \ell) \in \text{para}(P)$ , and
- (2)  $|J| + \ell \leq g(k)$  for some computable function  $g$ .

It can be verified that a problem  $P$  is FPT if and only if the problem  $\text{para}(P)$  admits a kernelization. If  $g$  is a polynomial function, then  $\text{para}(P)$  is said to admit a *polynomial* kernelization.

In some cases, it is possible that a parameterized problem  $\text{para}(P)$  is preprocessed into an equivalent instance of a different problem satisfying some special properties. Such preprocessing algorithms are also well-known in the literature. A *polynomial compression* of a parameterized problem  $\text{para}(P)$  into a problem  $Q$  is an algorithm that given an instance  $(I, k)$  of  $\text{para}(P)$ , runs in time polynomial in  $|I| + k$  time, and outputs an instance  $J$  of  $Q$  such that

- (1)  $(I, k)$  is a YES-instance of  $\text{para}(P)$  if and only if  $J$  is a YES-instance of  $Q$ , and
- (2)  $|J| \leq f(k)$  for a polynomial function  $f$ .

For a detailed introduction to parameterized complexity and related terminologies, we refer the reader to the recent books by Cygan et al. [76], by Downey and Fellows [82] and by Fomin et al. [86].

Next, we introduce some of the structural graph parameters that we have studied in this thesis.

## Treewidth

One of the most well-studied structural parameters is *treewidth* (which, informally, quantifies how close the input graph is to a tree, and is denoted by  $\text{tw}$ ). We begin with what is called a *tree decomposition* of a graph.

**Definition 2.1.** A *tree decomposition* of an undirected graph  $G = (V, E)$  is a pair  $\mathcal{T} = (T, \mathcal{X} = \{X_t\}_{t \in V(T)})$ , where  $T$  is a tree and  $\mathcal{X}$  is a collection of subsets of  $V(G)$  such that

- (1) for every vertex  $u \in V(G)$ , there is  $t \in V(T)$  such that  $u \in X_t$ ,
- (2) for every edge  $uv \in E(G)$ , there is  $t \in V(T)$  such that  $u, v \in X_t$ , and
- (3) for every vertex  $u \in V(G)$ , the set of nodes in  $T$  that contains  $u$ , that is,  $\{t \in V(T) \mid u \in X_t\}$  forms a connected subgraph of  $T$ .

For each  $t \in V(T)$ , the set  $X_t$  is called a *bag* of  $V(G)$  corresponding to the node  $t$ . Given a tree decomposition  $\mathcal{T}$ , the *width* of  $\mathcal{T}$  is defined as

$$\max_{t \in V(T)} \{|X_t| - 1\}$$

and the *treewidth* of a graph is the minimum width over all possible tree decompositions of  $G$ . See Figure 2.12 for an example of tree decomposition and treewidth of a 5-cycle. We shall denote the treewidth of a graph  $G$  by  $\text{tw}(G)$ , or simply by  $\text{tw}$ , when the graph  $G$  is understood from the context. Trivially, any graph with  $n$  vertices has treewidth at most  $n - 1$ . Informally, treewidth is a measure of how close a graph is to a tree. Moreover, the treewidth of a tree is 1.

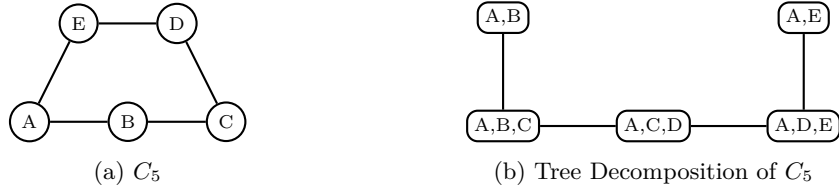


Figure 2.12: Example of tree decomposition of a 5-cycle. The rectangular vertices in (b) represent the bags of the decomposition. In this case, the treewidth is 2 (which is also true for all  $n$ -cycles).

We also refer readers to [76, Chapter 7] for further notes on treewidth. Courcelle's celebrated theorem [74] states that the class of graph problems expressible in Monadic Second-Order Logic (MSOL) by a formula of constant size admit an algorithm running in time  $f(\text{tw}) \cdot \text{poly}(n)$ . Hence, a large class of problems admit an FPT algorithm when parameterized by the treewidth. Unfortunately, the function  $f$  can often be a tower of exponents whose height depends roughly on the size of the MSOL formula. Hence, studying graph problems parameterized by treewidth and designing algorithms for them are also of much interest in their own right.

## Vertex cover number

A *vertex cover* of a graph  $G$  is a vertex subset  $S$  of  $G$  such that for each edge  $uv$  of  $G$ , either  $u \in S$  or  $v \in S$ . The *vertex cover number* of a graph  $G$  is the minimum of the cardinalities of all vertex covers of  $G$ . See Figure 2.13 for an example of a graph with vertex cover number 4. We shall denote the vertex cover number of a graph  $G$  by  $\text{vc}(G)$ , or simply by the notation<sup>1</sup>  $\text{vc}$ , when the graph  $G$  is clear from the context. In relation to treewidth, we have  $\text{tw} \leq \text{vc}$ . This is because, for any graph  $G$  and any vertex cover  $S$  of  $G$ , define a bag  $B_i$  by taking the vertices in  $S$  and a vertex  $v_i$  of the independent set  $V(G) \setminus S$ . Then, adding the edges  $B_i B_{i+1}$  for all  $v_i \in V(G) \setminus S$  gives a tree decomposition of  $G$  of width  $|S|$ .

## Neighborhood diversity

The *neighborhood diversity* of a graph  $G$  is the smallest integer  $\text{ndv} = \text{ndv}(G)$  such that  $G$  can be partitioned into  $\text{ndv}$  sets of mutual twin vertices, each set being either a clique or an independent set.

<sup>1</sup>Since each graph is also a hypergraph (to be defined in detail in Chapter 3) with each hyperedge being of order 2, the vertex cover number is also sometimes denoted by  $\tau(G)$  which usually denotes the *covering number* of a hypergraph

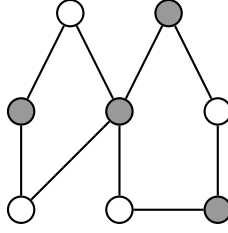


Figure 2.13: Example of a graph  $G$  with vertex cover number 4. The shaded vertices form a minimum-ordered vertex cover of  $G$ .

It was introduced by Lampis in [157]. See Figure 2.14 for an example of a graph with neighborhood diversity 4. With respect to vertex cover, we have  $\text{ndv} \leq 2^{\text{vc}} + \text{vc}$  since, given any vertex cover  $S$  of a graph  $G$  partitions the set  $V(G) \setminus S$  into at most  $2^{|S|}$  parts such that in each part, the vertices are pairwise open-twins in  $G$ . Moreover, there can be at most  $|S|$  such parts in  $S$ . This implies that  $\text{ndv} \leq 2^{|S|} + |S|$ .

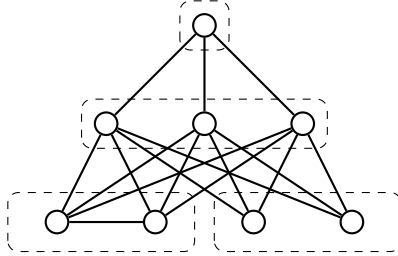


Figure 2.14: Example of a graph with neighborhood diversity 4. The vertices inside the dashed rectangles either form cliques or independent sets.

### Twin-cover number

A *twin-cover* of a graph  $G$  is a vertex subset of  $G$  such that, for any edge  $uv \in E(G)$  such that  $u \notin S$  and  $v \notin S$ , the pair  $u, v$  are closed twins in  $G$ . In other words, the set  $V(G) \setminus S$  can be partitioned such that each part is a clique and any two vertices in a part are closed twins in  $G$ . Moreover, the *twin-cover number* of a graph is the minimum possible order of a twin cover of  $G$ . It was first defined by Ganian in [108]. See Figure 2.15 for an example of a graph with twin-cover number 2. We denote the twin-cover number of a graph  $G$  by  $\text{tc}(G)$ , or simply by  $\text{tc}$ , when the graph  $G$  is clear from the context. The twin cover number is at most the vertex cover number, since the vertices not in the vertex cover form cliques of order 1 in the remaining graph.

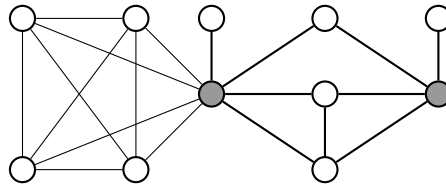


Figure 2.15: Example of a graph  $G$  with twin-cover number 2. The shaded vertices form a minimum-ordered twin-cover of  $G$ .

### Distance to clique

The *distance to clique* of a graph is the order of a smallest set of vertices that need to be removed from a graph so that the remaining vertices form a clique. See Figure 2.16 for an example of a graph with distance to clique 2. We denote the distance to clique of a graph  $G$  by  $\text{dc}(G)$ , or simply by  $\text{dc}$ , when the graph  $G$  is well-understood from the context. It is a dense analogue of the vertex cover number (which can be seen as the “distance to independent set”). Thus, distance to clique is the same as the vertex cover number in the complement graph. Moreover, we have  $\text{ndv} \leq 2^{\text{dc}} + \text{dc}$ . This is for very similar reasons as the relation between  $\text{ndv}$  and  $\text{vc}$ , only this time, for any set  $S$  of vertices of a graph  $G$  such that  $V(G) \setminus S$  is a clique, the latter can be partitioned into at most  $2^{|S|}$  parts, in each of which, the vertices are pairwise closed-twins in  $G$ .

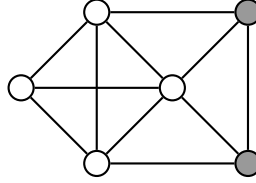


Figure 2.16: Example of a graph  $G$  with distance to clique 2. The shaded vertices form a minimum-ordered set, removing which, the remaining graph is complete ( $K_4$ ).

### Feedback edge set number

An edge subset  $X$  of a graph  $G$  is called a *feedback edge set* of  $G$  if the graph  $G - X$  is a forest. Moreover, the minimum order of a feedback edge set of  $G$  is known as the *feedback edge set number* of  $G$ . See Figure 2.17 for an example of a graph with feedback edge set number 4. This parameter of  $G$  is denoted by  $\text{fes}(G)$ , or simply by  $\text{fes}$  when the graph  $G$  is clear from the context. We note here that  $\text{tw} \leq \text{fes} + 1$ . This is because, on any graph  $G$  with a feedback edge set  $S$ , the subgraph  $G - S$  is a forest with components (trees), say  $T_i$ . Each  $T_i$  has a tree decomposition, say  $\mathcal{T}_i$  with bags  $B_i$  of width at most 2. Update each such bag  $B_i$  to  $B_i \cup \{u, v : uv \in S\}$ . Finally, add edges between the  $\mathcal{T}_i$ 's so that the final graph is a tree  $\mathcal{T}$ . The latter is a tree decomposition of  $G$  with bags of size at most  $|S| + 2$  and hence,  $\text{tw} \leq |S| + 1$ .

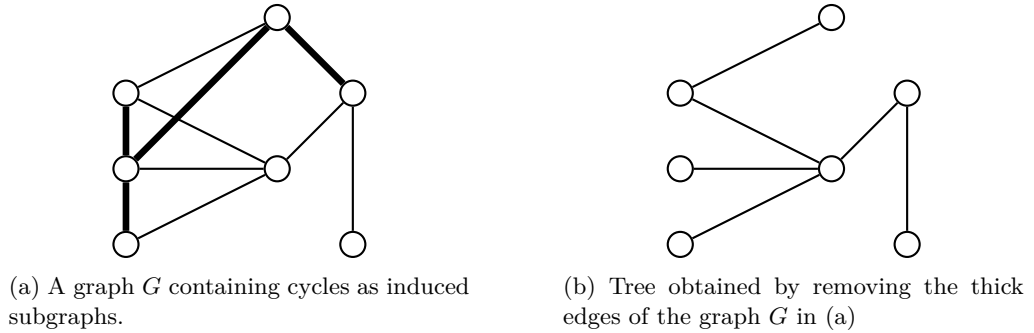


Figure 2.17: Example of a graph with feedback edge set number 4. The thick edges of the graph  $G$  in (a) form a minimum-ordered edge subset of  $G$  whose removal renders the graph a tree.

## 2.2 Separating sets revisited

In this section, we revisit the separation properties of the various codes that we had introduced in Definitions 1.1 – 1.4.

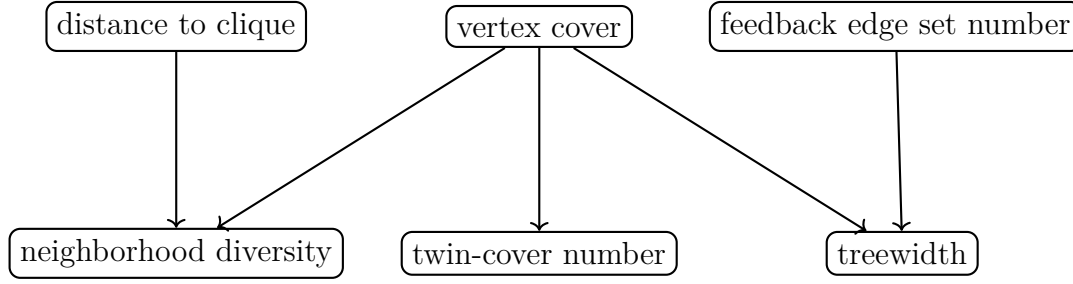


Figure 2.18: Schematic representation of structural parameters of graphs considered in this thesis. The arrow from a parameter  $q$  to a parameter  $p$  represents that  $p \leq f(q)$  for some computable function  $f$ .

## 2.2.1 Twins and separators

In Chapter 1, we have encountered the concepts of (closed or open) twins and twin-free graphs. In this section, we redefine these concepts mostly formulating equivalent definitions in terms of newer concepts, terminologies and notations around them.

Recall that any two vertices  $u, v$  of a graph  $G$  are called *open twins* (of each other in  $G$ ) if  $N_O(G; u) = N_O(G; v)$ . See Figure 2.19a for an example. Let  $\Delta_O(G; u, v) = N_O(G; u) \Delta N_O(G; v)$  be called the  $(u, v)$ -*O-separator*. Then equivalently, any two distinct vertices  $u, v$  of  $G$  are open twins if and only if  $\Delta_O(G; u, v) = \emptyset$ . A graph  $G$  is called *open-twin-free* if  $G$  has no pair of distinct vertices which are open twins. Then, the following remark provides an alternative definition of an open-twin-free graph.

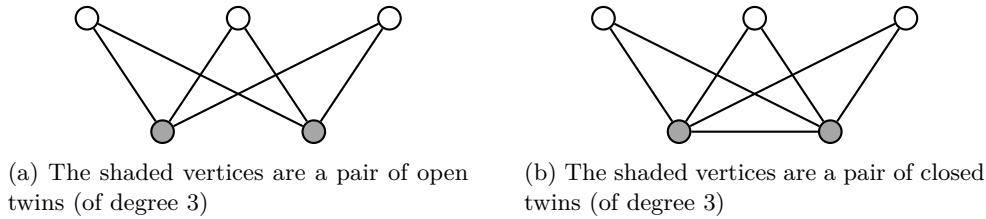


Figure 2.19: Examples of twins. The pair of gray vertices represent a pair of twins.

**Remark 2.7.** A graph  $G$  is open-twin-free if and only if  $\Delta_O(G; u, v) \neq \emptyset$  for all distinct vertices  $u, v \in V(G)$ .

Also recall that any two vertices  $u, v$  of a graph  $G$  are called *closed twins* (of each other in  $G$ ) if  $N_C(G; u) = N_C(G; v)$ . See Figure 2.19b for an example. Let  $\Delta_C(G; u, v) = N_C(G; u) \Delta N_C(G; v)$  be called the  $(u, v)$ -*C-separator*. Then alternatively, any two distinct vertices  $u, v$  of  $G$  are closed twins if and only if  $\Delta_C(G; u, v) = \emptyset$ . A graph  $G$  is called *closed-twin-free* if  $G$  has no pair of distinct vertices which are closed twins. Then, the following remark provides an equivalent definition of a closed-twin-free graph.

**Remark 2.8.** A graph  $G$  is closed-twin-free if and only if  $\Delta_C(G; u, v) \neq \emptyset$  for all distinct vertices  $u, v \in V(G)$ .

Whenever vertices  $u, v$  of a given graph  $G$  and the separation types (O and C) are well-understood from the context, we may often call a  $(u, v)$ -O-separator (respectively, a  $(u, v)$ -C-separator) by either a  $(u, v)$ -separator or an O-separator (respectively, a C-separator) or just a separator.

**Observation 2.2.** *If  $G$  is a graph and  $u, v$  are two distinct vertices of  $G$ , then we have*

$$\Delta_C(G; u, v) = \begin{cases} \Delta_O(G; u, v) \cup \{u, v\}, & \text{if } uv \notin E(G); \\ \Delta_O(G; u, v) \setminus \{u, v\}, & \text{if } uv \in E(G). \end{cases}$$

## 2.2.2 Separation of vertices

In Chapter 1, we have encountered four types of separation properties, namely location, open separation, closed separation and full separation, in the definitions of the codes introduced there. In this section, we address them again from the point of view of *separators*.

### 2.2.2.1 Open separation

A vertex  $w$  of a graph  $G$  is said to *open-separate* (or *O-separate*) a pair  $u, v$  of distinct vertices of  $G$  if  $w \in \Delta_O(G; u, v)$ . Moreover, if the set  $\Delta_O(G; u, v) = \{w\}$  is a singleton, then the vertex  $w$  is called *open-separation-forced* (or *O-separation forced* or simply *forced*) with respect to  $u$  and  $v$ . Any vertex subset  $S$  of  $G$  is said to *open-separate* (or *O-separate*) the pair  $u, v$  if there exists a vertex  $w \in S$  such that  $w$  open-separates  $u, v$ , that is, if  $\Delta_O(G; u, v) \cap S \neq \emptyset$ . Moreover,  $S$  is said to be an *open-separating set* (or an *O-separating set*) of the graph  $G$  if  $S$  open-separates every pair of distinct vertices of  $G$ , that is, if  $\Delta_O(G; u, v) \cap S \neq \emptyset$  for all pairs of distinct vertices  $u, v \in V(G)$ .

With the above definitions, notice that being an open-separating set is equivalent to possessing the property of open separation as introduced in Definitions 1.3 and 1.7 in Chapter 1. Moreover, a graph  $G$  is said to be *open-separable* (or *O-separable*) if there exists an open-separating set of  $G$ . Then, notice that, in order for a graph  $G$  to be open-separable, we must have  $\Delta_O(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . Conversely, if  $\Delta_O(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ , it implies that the vertex set  $V(G)$  is an open-separating set of  $G$  and hence, the latter is open-separable.

We now summarize these concepts in the following remarks.

**Remark 2.9.** *A graph  $G$  is open-separable if and only if  $\Delta_O(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ , that is, if and only if  $G$  is open-twin-free. In particular, any open-separable graph has at most one isolated vertex.*

**Remark 2.10.** *An OD-code of a graph  $G$  exists if and only if  $G$  is open-separable.*

*Proof.* By Remark 2.2, the graph  $G$  always has a dominating set. Moreover, by definition,  $G$  has an open-separating set if and only if  $G$  is open-separable.  $\square$

**Remark 2.11.** *An OTD-code of a graph  $G$  exists if and only if  $G$  is open-separable and isolate-free.*

*Proof.* By Remark 2.3,  $G$  has a total-dominating set if and only if  $G$  is isolate-free. Moreover, by definition,  $G$  has an open-separating set if and only if  $G$  is open-separable.  $\square$

See Figures 1.7c and 1.5 in Chapter 1 for examples of an OD-code and an OTD-code, respectively.

### 2.2.2.2 Closed separation

A vertex  $w$  of  $G$  is said to *closed-separate* (or *C-separate*) a pair  $u, v$  of distinct vertices of  $G$  if  $w \in \Delta_C(G; u, v)$ . Moreover, if the set  $\Delta_C(G; u, v) = \{w\}$  is a singleton, then the vertex  $w$  is called *closed-separation-forced* (or *C-separation forced* or simply *forced*) with respect to  $u$  and  $v$ . Any vertex subset  $S$  of  $G$  is said to *closed-separate* the pair  $u, v$  if there exists a vertex  $w \in S$  such that  $w$  closed-separates  $u, v$ , that is, if  $\Delta_C(G; u, v) \cap S \neq \emptyset$ . Moreover,  $S$  is said to be a *closed-separating set* (or a *C-separating set*) of the graph  $G$  if  $S$  closed-separates every pair of distinct vertices of  $G$ , that is, if  $\Delta_C(G; u, v) \cap S \neq \emptyset$  for all pairs of distinct vertices  $u, v \in V(G)$ .

From the above definitions, we notice that being a closed-separating set is equivalent to possessing the property of closed separation as introduced in Definitions 1.6 and 1.2 in Chapter 1. Moreover, a graph  $G$  is said to be *closed-separable* (or *C-separable*) if there exists a closed-separating set of  $G$ . Thus, we notice that, in order for a graph  $G$  to be closed-separable, we must have  $\Delta_C(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . Conversely, if  $\Delta_C(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ , it implies that the vertex set  $V(G)$  is a closed-separating set of  $G$  and hence, the latter is closed-separable.

We now summarize these concepts in the following remarks.

**Remark 2.12.** *A graph  $G$  is closed-separable if and only if  $\Delta_C(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ , that is, if and only if  $G$  is closed-twin-free.*

**Remark 2.13.** *An ID-code of a graph  $G$  exists if and only if  $G$  is closed-separable.*

*Proof.* By Remark 2.2,  $G$  always has a dominating set. In addition, by definition,  $G$  has a closed-separating set if and only if  $G$  is closed-separable.  $\square$

**Remark 2.14.** *An ITD-code of a graph  $G$  exists if and only if  $G$  is closed-separable and isolate-free.*

*Proof.* By Remark 2.3,  $G$  has a total-dominating set if and only if  $G$  is isolate-free. Moreover, by definition,  $G$  has a closed-separating set if and only if  $G$  is closed-separable.  $\square$

See Figures 1.4 and 1.7b in Chapter 1 for examples of an ID-code and an ITD-code, respectively.

### 2.2.2.3 Location

For any two distinct vertices  $u$  and  $v$  of a graph  $G$ , let  $\Delta_L(G; u, v) = \Delta_O(G; u, v) \cup \{u, v\}$  be called the  $(u, v)$ -*L-separator*. Notice that  $\Delta_C(G; u, v) \cup \{u, v\} = \Delta_O(G; u, v) \cup \{u, v\}$  for all distinct  $u, v \in V(G)$ . Therefore, we may also define the  $(u, v)$ -L-separator to be the set  $\Delta_C(G; u, v) \cup \{u, v\}$ . Whenever the vertices  $u, v$  and the separation type (L) are well-understood from the context, we may often call a  $(u, v)$ -L-separator to be either a  $(u, v)$ -separator or an *L-separator* or just a *separator*. In contrast to closed separation and open separation, whereby the respective separators may be singletons thus giving rise to forced vertices, the L-separators are never singletons and hence, there can never be any “location-forced vertices”. A vertex  $w$  of  $G$  is said to *locate* (or *L-separate*) a pair  $u, v$  of distinct vertices of  $G$  if  $w \in \Delta_L(G; u, v)$ . Any vertex subset  $S$  of  $G$  is said to *locate* (or *L-separate*) the pair  $u, v$  if there exists a vertex  $w \in S$  such that  $w$  locates  $u, v$ , that is, if  $\Delta_L(G; u, v) \cap S \neq \emptyset$ . A vertex subset  $S$  of a graph  $G$  is said to be a *locating set* (or an *L-separating set*) of  $G$  if  $S$  locates every pair of distinct vertices of  $G$ , that is, if  $\Delta_L(G; u, v) \cap S \neq \emptyset$  for all pairs of distinct vertices  $u, v \in V(G)$ .

**Remark 2.15.** *An LD-code of a graph always exists.*

*Proof.* Let  $G$  be a graph and let  $u, v$  be any two distinct vertices of  $G$ . Then by definition, we have  $\Delta_L(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . This implies that we can always choose a vertex subset  $S$  of  $G$  such that  $\Delta_L(G; u, v) \cap S \neq \emptyset$ . Therefore,  $S$  is a locating set of  $G$ . In other words, a locating set of any graph always exists and hence, the result.  $\square$

The following observation has been made by Argiroffo et al. in [11].

**Observation 2.3.** *If  $G$  is a graph and  $u, v$  are two distinct vertices of  $G$ , then we have*

$$\Delta_L(G; u, v) = \begin{cases} \Delta_C(G; u, v), & \text{if } uv \notin E(G), \\ \Delta_O(G; u, v), & \text{if } uv \in E(G). \end{cases}$$

*Therefore, in particular,  $\Delta_L(G; u, v) \in \{\Delta_C(G; u, v), \Delta_O(G; u, v)\}$  for all distinct  $u, v \in V(G)$ .*

The following observation shows that being a locating set is equivalent to possessing the property of location introduced in Definitions 1.1 and 1.5 in Chapter 1.



**Observation 2.4.** *Let  $G$  be a graph and let  $S$  be a vertex subset of  $G$ . Then,  $S$  is a locating set of  $G$  if and only if  $\Delta_O(G; u, v) \cap S \neq \emptyset$  for all distinct vertices  $u, v \in V(G) \setminus S$ .*

*Proof.* Let us first assume that  $S$  is a locating set of  $G$ . Then, we have  $\Delta_L(G; u, v) \cap S \neq \emptyset$  for all distinct vertices  $u, v \in V(G)$ . Let  $u, v \in V(G) \setminus S$ . Let us first assume that  $uv \in E(G)$ . Then, by Observation 2.3, we have  $\Delta_L(G; u, v) \cap S = \Delta_O(G; u, v) \cap S$ . On the other hand, if  $uv \notin E(G)$ , then using Observations 2.2 and 2.3, we have  $\Delta_L(G; u, v) \cap S = (\Delta_O(G; u, v) \cup \{u, v\}) \cap S = \Delta_O(G; u, v) \cap S$  (the last equality follows by the assumption that  $u, v \notin S$ ). Hence, we have  $\Delta_L(G; u, v) \cap S = \Delta_O(G; u, v) \cap S$  for all distinct  $u, v \in V(G) \setminus S$ . Since, by assumption  $\Delta_L(G; u, v) \cap S \neq \emptyset$  for all distinct  $u, v \in V(G)$ , it implies that  $\Delta_O(G; u, v) \cap S \neq \emptyset$  for all distinct  $u, v \in V(G) \setminus S$ . This proves the necessary part of the result.

Let us now assume that  $\Delta_O(G; u, v) \cap S \neq \emptyset$  for all distinct vertices  $u, v \in V(G) \setminus S$ . Let  $u, v \in V(G)$  be distinct. If  $u, v \notin S$ , then we have  $\Delta_O(G; u, v) \cap S = (N_O(G; u) \cap S) \triangle (N_O(G; v) \cap S) \neq \emptyset$  (the last inequality follows by assumption). Let us therefore assume, that  $\{u, v\} \cap S \neq \emptyset$ . Since  $\{u, v\} \subseteq \Delta_L(G; u, v)$  by definition, it implies that  $\Delta_L(G; u, v) \cap S \neq \emptyset$ . This proves the sufficiency part of the result.  $\square$

**Remark 2.16.** *An LD-code of a graph  $G$  always exists.*

*Proof.* Since  $G$  always has a dominating set, say  $D$  and, by Remark 2.15, always has a locating set, say  $S$ , the set  $D \cup S$  is an LD-code of  $G$ .  $\square$

**Remark 2.17.** *An LTD-code of a graph  $G$  exists if and only if  $G$  is isolate-free.*

*Proof.* By Remark 2.15,  $G$  always has a locating set and, by Remark 2.3,  $G$  has a total-dominating set  $T$  if and only if  $G$  is isolate-free.  $\square$

See Figures 1.3 and 1.7a in Chapter 1 for examples of an LD-code and an LTD-code, respectively.

### 2.2.2.4 Full separation

For any two distinct vertices  $u$  and  $v$  of a graph  $G$ , let  $\Delta_F(G; u, v) = \Delta_O(G; u, v) \setminus \{u, v\}$  be called the  $(u, v)$ -*F-separator*. Moreover, if the set  $\Delta_C(G; u, v) = \{w\}$  is a singleton, then the vertex  $w$  is called *full-separation-forced* (or *F-separation forced* or simply *forced*) with respect to  $u$  and  $v$ . Notice that  $\Delta_C(G; u, v) \setminus \{u, v\} = \Delta_O(G; u, v) \setminus \{u, v\}$  for all distinct  $u, v \in V(G)$ . Therefore, we may also define the  $(u, v)$ -F-separator to be the set  $\Delta_C(G; u, v) \setminus \{u, v\}$ . Whenever vertices  $u, v$  and the separation type (F) are well-understood from the context, we may often call a  $(u, v)$ -F-separator by either a  $(u, v)$ -separator or an *F-separator* or just a *separator*. A vertex  $w$  of  $G$  is said to *full-separate* (or *F-separate*) a pair  $u, v$  of distinct vertices of  $G$  if  $w \in \Delta_F(G; u, v)$ . Any vertex subset  $S$  of  $G$  is said to *full-separate* the pair  $u, v$  if there exists a vertex  $w \in S$  such that  $w$  full-separates  $u, v$ , that is, if  $\Delta_F(G; u, v) \cap S \neq \emptyset$ . A vertex subset  $S$  of a graph  $G$  is said to be a *full-separating set* (or an *F-separating set*) of  $G$  if  $S$  full-separates every pair of distinct vertices of  $G$ , that is, if  $\Delta_F(G; u, v) \cap S \neq \emptyset$  for all pairs of distinct vertices  $u, v \in V(G)$ .

Now, analogous to Observation 2.3 in the case of location, we have the following for full-separating sets.

**Observation 2.5** ([53]). *If  $G$  is a graph and  $u, v$  are two distinct vertices of  $G$ , then we have*

$$\Delta_F(G; u, v) = \begin{cases} \Delta_O(G; u, v), & \text{if } uv \notin E(G), \\ \Delta_C(G; u, v), & \text{if } uv \in E(G). \end{cases}$$

*Therefore, in particular,  $\Delta_F(G; u, v) \in \{\Delta_C(G; u, v), \Delta_O(G; u, v)\}$  for all distinct  $u, v \in V(G)$ .*

*Proof.* Let us first assume that  $uv \notin E(G)$ . Since  $u \notin N_G(u)$  and  $v \notin N_G(v)$ , we therefore have  $\Delta_O(G; u, v) \cap \{u, v\} = \emptyset$ . This implies that  $\Delta_O(G; u, v) = \Delta_O(G; u, v) \setminus \{u, v\}$  and hence,  $\Delta_F(G; u, v) = \Delta_O(G; u, v)$ . Let us now assume that  $uv \in E(G)$ . In this case, by Observation 2.2, we have  $\Delta_O(G; u, v) \setminus \{u, v\} = \Delta_C(G; u, v)$  and therefore,  $\Delta_F(G; u, v) = \Delta_C(G; u, v)$ .  $\square$

The next observation proves that being a full-separating set is equivalent to possessing the property of full separation as introduced in Definitions 1.8 and 1.4 in Chapter 1.

**Observation 2.6** ([53]). *Let  $G$  be a graph and let  $S$  be a vertex subset of  $G$ . Then,  $S$  is a full-separating set of  $G$  if and only if  $\Delta_O(G; u, v) \cap S \neq \emptyset$  and  $\Delta_C(G; u, v) \cap S \neq \emptyset$  for all distinct vertices  $u, v \in V(G)$ .*

*Proof.* We first prove the necessary part. So, let us first assume that  $S$  is a full-separating set of  $G$ . Then, we have  $\Delta_F(G; u, v) \cap S \neq \emptyset$  for all distinct vertices  $u, v \in V(G)$ . We first show that  $\Delta_O(G; u, v) \cap S \neq \emptyset$  for all distinct  $u, v \in V(G)$ . So, let  $u, v$  be two distinct vertices of  $G$ . If  $uv \notin E(G)$ , by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_O(G; u, v)$ . Now, since  $\Delta_F(G; u, v) \cap S \neq \emptyset$  by assumption, we also have  $\Delta_O(G; u, v) \cap S \neq \emptyset$ . Let us now assume that  $uv \in E(G)$ . Then, by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_C(G; u, v)$ . Again, by assumption, we have  $\Delta_F(G; u, v) \cap S \neq \emptyset$  and hence, we also have  $\Delta_C(G; u, v) \cap S \neq \emptyset$ . Since,  $uv \in E(G)$ , by Observation 2.2, we have  $\Delta_C(G; u, v) \subset \Delta_O(G; u, v)$ . This implies that  $\Delta_O(G; u, v) \cap S \neq \emptyset$  as well.

We now show that  $\Delta_C(G; u, v) \cap S \neq \emptyset$  for all distinct  $u, v \in V(G)$ . So, let  $u, v$  be two distinct vertices of  $G$ . If  $uv \in E(G)$ , by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_C(G; u, v)$ . Now, since  $\Delta_F(G; u, v) \cap S \neq \emptyset$  by assumption, we also have  $\Delta_C(G; u, v) \cap S \neq \emptyset$ . Let us now assume that  $uv \notin E(G)$ . Then, by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_O(G; u, v)$ . Again, by assumption, we have  $\Delta_F(G; u, v) \cap S \neq \emptyset$  and hence, we also have  $\Delta_O(G; u, v) \cap S \neq \emptyset$ . Since  $uv \notin E(G)$ , by Observation 2.2, we have  $\Delta_O(G; u, v) \subset \Delta_C(G; u, v)$ . This implies that  $\Delta_C(G; u, v) \cap S \neq \emptyset$  as well. This proves the necessary part of the result.

We now prove the sufficiency part of the result. So, let us assume that  $\Delta_O(G; u, v) \cap S \neq \emptyset$  and  $\Delta_C(G; u, v) \cap S \neq \emptyset$  for all distinct vertices  $u, v \in V(G)$ . Also, let  $u, v \in V(G)$  be distinct. Let us first assume that  $uv \notin E(G)$ . Then, by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_O(G; u, v)$ . Since by assumption,  $\Delta_O(G; u, v) \cap S \neq \emptyset$ , we therefore have,  $\Delta_F(G; u, v) \cap S \neq \emptyset$ . Let us now assume that  $uv \in E(G)$ . Then, again by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_C(G; u, v)$ . Again, by assumption, we have  $\Delta_C(G; u, v) \cap S \neq \emptyset$ . Hence,  $\Delta_F(G; u, v) \cap S \neq \emptyset$  as well.  $\square$

A graph  $G$  is said to be *full-separable* (or *F-separable*) if there exists a full-separating set of  $G$ . Then, by definition, in order for  $G$  to be full-separable, we must have  $\Delta_F(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . Conversely, if  $\Delta_F(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ , it implies that the vertex set  $V(G)$  is a full-separating set of  $G$  and hence, the latter is full-separable.

We therefore summarize these ideas in the next remarks.

**Remark 2.18** ([53]). *A graph  $G$  is full-separable if and only if  $\Delta_F(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ , that is, if and only if  $G$  is twin-free.*

*Proof.* The first if and only if part of the result follows immediately. Let us therefore prove that  $\Delta_F(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$  if and only if  $G$  is twin-free. We first show the necessary part. So, let us assume that  $\Delta_F(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . If on the contrary, a pair of open twins  $u, v \in V(G)$  exist, then we have  $uv \notin E(G)$ . Then, by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_O(G; u, v)$ . Hence, using our assumption, we have  $\Delta_O(G; u, v) \neq \emptyset$  thus contradicting the fact that  $u$  and  $v$  are open twins. On the other hand, if on the contrary, a pair of closed twins  $u, v \in V(G)$  exist, then we have  $uv \in E(G)$ . Then, by Observation 2.5, we have  $\Delta_F(G; u, v) = \Delta_C(G; u, v)$ . Again, using our assumption, we have  $\Delta_C(G; u, v) \neq \emptyset$  thus contradicting the fact that  $u$  and  $v$  are closed twins. This proves that  $G$  is twin-free.

We now prove the sufficiency part of the result. So, let us assume that  $G$  is twin-free. This implies that both  $\Delta_C(G; u, v) \neq \emptyset$  and  $\Delta_O(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . Therefore, by Observation 2.5 again, we have  $\Delta_F(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ .  $\square$

**Remark 2.19.** *An FD-code of a graph  $G$  exists if and only if  $G$  is full-separable.*

*Proof.* By Remark 2.2, the graph  $G$  always has a dominating set. In addition, by definition,  $G$  has a full-separating set if and only if  $G$  is full-separable.  $\square$

**Remark 2.20.** An FTD-code of a graph  $G$  exists if and only if  $G$  is full-separable and isolate-free.

*Proof.* By Remark 2.3,  $G$  has a total-dominating set if and only if  $G$  is isolate-free. Moreover, by definition,  $G$  has a full-separating set if and only if  $G$  is full-separable.  $\square$

See Figures 1.7d and 1.6 in Chapter 1 for examples of an FD-code and an FTD-code, respectively.

## 2.3 A unified system of notations for codes

Recall that we had defined  $\text{NBD-TYPE} = \{C, O\}$  to abbreviate the neighborhood types of vertices of graphs. Similarly, we now introduce the set  $\text{SEP-TYPE} = \{C, O, L, F\}$  to denote the set of abbreviations of all the four separation types that we have introduced so far. Also, recall from Chapter 1 that  $\text{CODES} = \{LD, LTD, ID, ITD, OD, OTD, FD, FTD\}$  and, for any  $X \in \text{CODES}$ , if an  $X$ -code of a graph  $G$  exists, then  $G$  is called  $X$ -admissible. Moreover, for any  $X \in \text{CODES}$ ,  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$ , by the notation  $X \equiv (A, B)$ , we shall mean from here on that any  $X$ -code of an  $X$ -admissible graph  $G$  is an  $A$ -separating set and a  $B$ -dominating set of  $G$ . Again from Chapter 1, we recall the following quantity to be the  $X$ -number of an  $X$ -admissible graph  $G$ .

$$\gamma^X(G) = \min\{|S| : S \subseteq V(G) \text{ and } S \text{ is an } X\text{-code of } G\}.$$

Whenever the element  $X \in \text{CODES}$  is well-understood from the context, we may refer to an  $X$ -number as just a *code number*. Any  $X$ -code  $S$  of  $G$  such that  $|S| = \gamma^X(G)$  is called a *minimum  $X$ -code* or an *optimal  $X$ -code* of  $G$ . Using Observations 2.3 and 2.5, we have the following remark.

**Remark 2.21.** Let  $G$  be a graph and let  $u, v \in V(G)$  be distinct. Moreover, let  $A \in \text{SEP-TYPE}$ . Then,  $\Delta_A(G; u, v) = \Delta_{T(A; u, v)}(G; u, v)$ , where  $T : \text{SEP-TYPE} \times V(G) \times V(G) \rightarrow \text{NBD-TYPE}$  is such that

$$T(A; u, v) = \begin{cases} C, & \text{if } A = C; \\ O, & \text{if } A = O; \\ O, & \text{if } A = L, uv \in E(G); \\ C, & \text{if } A = L, uv \notin E(G); \\ C, & \text{if } A = F, uv \in E(G); \\ O, & \text{if } A = F, uv \notin E(G). \end{cases} \quad (2.1)$$

The function  $T$  defined in Remark 2.21 captures the interplay between the various separation types all defined by means of either  $C$ - or  $O$ -neighborhoods of pairs of vertices. We call the function  $T$  to be the *type transformation function*.

**Observation 2.7.** For any separating set  $S$  of a graph  $G$ , there exists at most one vertex of  $G$  not dominated by  $S$ .

*Proof.* Let  $A \in \text{SEP-TYPE}$  and let  $S$  be an  $A$ -separating set of  $G$ . If possible, let there exist two vertices, say  $u$  and  $v$ , of  $G$  not dominated by  $S$ . In other words, we have  $N_G[u] \cap S = N_G[v] \cap S = \emptyset$ . This also implies that  $N_G(u) \cap S = N_G(v) \cap S = \emptyset$ . Hence, for all  $A \in \text{SEP-TYPE}$ , by the definition of an  $A$ -separating set, we arrive at a contradiction. This implies the result.  $\square$

**Remark 2.22.** Let  $X \in \text{CODES}$  and let  $G$  be an  $X$ -admissible graph. Then, the vertex set  $V(G)$  is an  $X$ -code of  $G$ .

## 2.4 Other prevalent terminologies related to codes

We also comment here regarding some of the other terminologies prevalent in the literature of identification problems. We recall here that if  $S$  is a code of a graph  $G$ , then any element of  $S$  is called a *code vertex* of  $G$ . In the literature of identification problems, code vertices have also been referred to as *codewords* (see, for example, [101, 133]). For some  $A \in \text{SEP-TYPE}$ ,  $B \in \text{NBD-TYPE}$  and  $X \in \text{CODES}$  such that  $X \equiv (A, B)$ , given any  $X$ -code  $S$  of  $G$  and any two distinct vertices  $u$  and  $v$  of a graph  $G$ , any vertex of  $G$  in  $\Delta_A(G; u, v) \cap S$  is called an *A-separating S-codeword* of  $G$ , or simply a *separating S-codeword* when the separation type  $A \in \text{SEP-TYPE}$  is clear from the context. In other words, an  $A$ -separating  $S$ -codeword is an element of  $S$  that  $A$ -separates a pair of distinct vertices of  $G$ .

In the context of ID-codes (and also LD-codes), given a graph  $G$ , a vertex  $v$  of  $G$  and a vertex subset  $S$  of  $G$ , the set  $N_G[v] \cap S$  has quite often been referred to in the literature as  $I_G(S; v)$ , or simply as  $I(v)$  when  $S$  and  $G$  are clear from context (see [101, 133, 146] for example). The set  $I_G(S; v)$  is called the *I-set* (an abbreviation for *identifying set*) of  $v$  in  $G$  with respect to the set  $S \subseteq V(G)$ . In other words, in terms of  $I$ -sets, an ID-code of a closed-twin-free graph  $G$  is a dominating set  $S$  such that the  $I$ -set of each vertex of  $G$  with respect to  $S$  is unique. Notice that, if  $v \in V(G) \setminus S$ , then we have  $I(v) = I_G(S; v) = N_G[v] \cap S = N_G(v) \cap S$ . Therefore, the terminology of  $I$ -sets is also utilized in the context of LD-codes since a dominating set  $S$  of a graph  $G$  is an LD-code if and only if each vertex in  $V(G) \setminus S$  has a unique  $I$ -set with respect to  $S$ .

## 2.5 Literature survey

We now shift our focus to a brief literature survey on all the eight problems at hand, namely, that of finding the  $X$ -number for  $X \in \text{CODES}$ . From a combinatorial point of view, a fair amount of study has been done to establish bounds on these code numbers. Bounds on LD-numbers, for example, have been studied in [35, 59] and ID-numbers have been studied in [59, 90]. In both these sets of papers, extremal graphs with respect to these codes have been studied. OTD-codes have been studied in [89] establishing a full characterization of graphs (namely, half-graphs) with full vertex sets as OTD-codes. In [101], the authors study and characterize extremal examples of ITD-codes of graphs  $G$  with  $\gamma^{\text{ITD}}(G) \geq |V(G)| - 1$ . On the other hand, it is well-known that a general lower bound on the code numbers of graphs is usually given in terms of logarithms of the number of vertices of the graphs. Characterizations and extremal examples of graphs whose code numbers attain these lower bounds have also been studied for some specific codes. For example, in [186], Slater provides a characterization of graphs whose LD-numbers reach the logarithmic lower bound. In [164], Moncel studies the same in the context of ID-codes. In [128], Henning and Rad provide constructions of graphs whose LTD-numbers attain the extremal lower bounds.

Apart from these extremal examples, bounds on code numbers of graphs from even further restricted families have been studied. To bring out the rather rich literature of such studies, we present in the next subsection in a tabular format a brief literature survey of these results on bounds.

### 2.5.1 Bounds on code numbers in specific graph families

In what follows, in Tables 2.1 – 2.10, we present a list of lower and upper bounds on the  $X$ -numbers of various graph families either already known from the literature or have been presented in this thesis. All the graphs whose bounds are presented in these tables are connected graphs.

### 2.5.1.1 Known bounds on LD-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$\lfloor \log n \rfloor$ [186]	TIGHT [186]	$n - 1$ [186]	TIGHT [186]
Trees	$\lfloor \frac{n}{3} \rfloor + 1$ [185]	TIGHT [185]	$n - 1$ [186]	TIGHT [186]
Paths	$\frac{2n}{5}$ [24]	TIGHT [24]	$\frac{2n}{5}$ [24]	TIGHT [24]
Cycles	$\frac{2n}{5}$ [24]	TIGHT [24]	$\frac{2n}{5}$ [24]	TIGHT [24]
$r$ -regular	$\frac{2n}{r+3}$ [187]	OPEN	$n - 1$ [186]	TIGHT [186]
Interval	$\sqrt{2n + \frac{9}{4} - \frac{3}{2}}$ [102]	TIGHT [102]	$n - 1$ [186]	TIGHT [186]
Unit interval	$\frac{n+1}{3}$ [102]	TIGHT [102]	$n - 1$ [186]	TIGHT [186]
Permutation	$\sqrt{n + \frac{9}{4} - \frac{1}{2}}$ [102]	TIGHT [102]	$n - 1$ [186]	TIGHT [186]
Bipartite permutation	$\frac{n-2}{3}$ [102]	OPEN	$n - 1$ [186]	TIGHT [186]
Cographs	$\frac{n}{3}$ [102]	TIGHT [102]	$n - 1$ [186]	TIGHT [186]
Bipartite	$\lfloor \log n \rfloor$ [186]	TIGHT [51] <b>Prop 3.1</b>	$n - 1$ [186]	TIGHT [186]
Cobipartite	$\lfloor \log n \rfloor$ [186]	TIGHT [51] <b>Prop 3.1</b>	$n - 1$ [186]	TIGHT [186]
Split	$\lfloor \log n \rfloor$ [186]	TIGHT [51] <b>Prop 3.1</b>	$n - 1$ [186]	TIGHT [186]
Chordal	$\lfloor \log n \rfloor$ [186]	TIGHT [51] <b>Prop 3.1</b>	$n - 1$ [186]	TIGHT [186]
Block	$\frac{n+1}{3}$ [47] <b>Thm 4.4</b>	TIGHT [47] <b>Prop 4.3</b>	$n - 1$ [186]	TIGHT [186]

Table 2.1: Known bounds for LD-numbers in some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

### Upper bounds on LD-numbers of twin-free graphs

Apart from the general upper and lower bounds on LD-numbers presented in Table 2.1, there is another question on upper bounds, namely on *twin-free* graphs, that has received a fair amount of attention in the literature. Garijo et al. in [112] and Foucaud and Henning in [96] had conjectured the following regarding LD-numbers of twin-free graphs.

**Conjecture 2.2** ([112, 95]). *Let  $G$  be a twin-free and isolate-free graph on  $n$  vertices. Then, we have*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

Conjecture 2.2 has been of much interest to researchers as this upper bound draws a parallel with the classical result by Ore [170] which also states that a minimum dominating set of an isolate-free graph is at most half the number of its vertices. In Table 2.2, we provide a list of all the graph families in which this conjecture has been known to hold.

Twin-free graph family	$\frac{n}{2}$ -Upper bound	Tightness
4-cycle-free	YES [112]	OPEN
Girth $\geq 5$ and minimum degree $\geq 2$	YES [18]	OPEN
Split	YES [98]	TIGHT [98]
Co-bipartite	YES [98]	TIGHT [98]
Line	YES [97]	TIGHT [97]
Maximal outerplanar	YES [65]	TIGHT [65]
Block	YES [47] <b>[Thm 4.3]</b>	TIGHT [47] <b>[Prop 4.2]</b>
Subcubic without open twins of degree 1 and 2	YES [48] <b>[Thm 4.7]</b>	TIGHT [48] <b>[Prop 4.5]</b>

Table 2.2: List of known graph families for which the conjectured  $n$ -half upper bound in [112] holds. The references marked in red are those that have been presented in this thesis.

### 2.5.1.2 Known bounds on LTD-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$\lfloor \log n \rfloor$ [128]	TIGHT [128]	$n - 1$ [128]	TIGHT [128]
Paths	$\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{4} \rceil - \lfloor \frac{n}{4} \rfloor$ [123]	TIGHT [123]	$\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{4} \rceil - \lfloor \frac{n}{4} \rfloor$ [123]	TIGHT [123]
Cycles	$\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{4} \rceil - \lfloor \frac{n}{4} \rfloor$ [128]	TIGHT [128]	$\lfloor \frac{n}{2} \rfloor + \lceil \frac{n}{4} \rceil - \lfloor \frac{n}{4} \rfloor$ [128]	TIGHT [128]
Trees	$\frac{2}{5}(n + 1)$ [123]	TIGHT [123]	$n - 1$ [128]	TIGHT [128]
Claw-free cubic	$\lfloor \log n \rfloor$ [128]	OPEN	$\frac{n}{2}$ [127]	TIGHT [127]

$r$ -regular	$\left\lceil \frac{2n}{r+2} \right\rceil$ [162]	TIGHT [162]	$n - 1$ [128]	OPEN
Cobipartite	$\lfloor \log n \rfloor$ [128]	TIGHT [51] <b>[Prop 3.2]</b>	$n - 1$ [128]	TIGHT [128]
Split	$\lfloor \log n \rfloor$ [128]	TIGHT [51] <b>[Prop 3.2]</b>	$n - 1$ [128]	TIGHT [128]
Chordal	$\lfloor \log n \rfloor$ [128]	TIGHT [51] <b>[Prop 3.2]</b>	$n - 1$ [128]	TIGHT [128]

Table 2.3: Known bounds for LTD-numbers of some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

### Upper bounds on LTD-numbers of twin-free graphs

Much in line with the conjecture by Garijo et al. in [112] on LD-numbers, Foucaud and Henning in [96] conjectured that any isolate-free twin-free graph has its LTD-number at most two-thirds the order of the graph. Formally, we have the following.

**Conjecture 2.3** ([96]). *Let  $G$  be a twin-free and isolate-free graph on  $n$  vertices. Then, we have*

$$\gamma^{\text{LTD}}(G) \leq \frac{2n}{3}.$$

This upper bound also draws a parallel with the result by Cockayne et al. in [67] which also states that a minimum total-dominating set of an isolate-free graph is at most two-thirds the number of its vertices. In Table 2.4, we provide a list of all the graph families in which this conjecture is known to hold.

Twin-free graph family	$\frac{2n}{3}$ -Upper bound	Tightness
4-cycle-free	YES [96]	TIGHT [96]
Line graphs	YES [97]	TIGHT [97]
Block	YES [39] <b>[Thm 4.11]</b>	TIGHT [39] <b>[Prop 4.9]</b>
Cobipartite	YES (with upper bound $\frac{n}{2}$ ) [39] <b>[Thm 4.9]</b>	TIGHT [98]
Split	YES (with upper bound $< \frac{2n}{3}$ ) [39] <b>[Thm 4.10]</b>	TIGHT [39] <b>[Prop 4.8]</b>
Subcubic (with open twins allowed)	YES [39] <b>[Thm 4.12]</b>	TIGHT [39] <b>[Prop 4.10]</b>

Table 2.4: List of known graph families for which the conjectured  $n$ -two-thirds upper bound in [96] holds. The references marked in red are those that have been presented in this thesis.

## 2.5.1.3 Known bounds on ID-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$\lceil \log(n+1) \rceil$ [149]	TIGHT [164]	$n-1$ [23, 110]	TIGHT [87]
Bipartite	$\lceil \log(n+1) \rceil$ [149]	TIGHT [87]	$n-1$ [23, 110]	TIGHT [87]
Paths	$\lfloor \frac{n}{2} \rfloor + 1$ [24]	TIGHT [24]	$\lfloor \frac{2n}{5} \rfloor + 1$ [24]	TIGHT [24]
Cycles	$\lfloor \frac{n}{2} \rfloor$ [24]	TIGHT [24]	$\frac{n}{2}$ [24]	TIGHT [24]
Trees	$\frac{3(n+1)}{7}$ [25]	TIGHT [25]	$n-1$ [23, 110]	TIGHT [87]
Interval	$\sqrt{2n + \frac{1}{4}} - \frac{1}{2}$ [102]	TIGHT [102]	$n-1$ [23, 110]	TIGHT [87, 90]
Unit interval	$\frac{n+1}{2}$ [102]	TIGHT [102]	$n-1$ [23, 110]	TIGHT [87, 90]
Cographs	$\frac{n+2}{2}$ [102]	TIGHT [102]	$n-1$ [23, 110]	TIGHT [87, 90]
Permutation	$\sqrt{n+2}$ [102]	TIGHT [102]	$n-1$ [23, 110]	TIGHT [87, 90]
Bipartite permutation	$\frac{n-2}{3}$ [102]	OPEN [102]	$n-1$ [23, 110]	TIGHT [87, 90]
Line	$\frac{3\sqrt{2}}{4}\sqrt{n}$ [87]	TIGHT [87]	$n-1$ [23, 110]	TIGHT [87, 90]
Girth $\geq 5$ and minimum degree $\geq 2$	$\frac{n-2}{3}$ [102]	OPEN [102]	$n-1$ [23, 110]	TIGHT [87]
Induced claw-free	$\Theta(\log n)$ [149]	TIGHT [87]	$n-1$ [23, 110]	TIGHT [87, 90]
Split	$\lceil \log(n+1) \rceil$ [149]	TIGHT [Prop 3.3]	$n-1$ [23, 110]	TIGHT [87, 90]
Chordal	$\lceil \log(n+1) \rceil$ [149]	TIGHT [87]	$n-1$ [23, 110]	TIGHT [87, 90]
Planar	$\frac{n+10}{7}$ [149]	OPEN [87]	$n-1$ [23, 110]	TIGHT [87, 90]
Series-parallel	$\frac{n+3}{4}$ [149]	OPEN [87]	$n-1$ [23, 110]	TIGHT [87, 90]
Outerplanar	$\frac{2n+3}{7}$ [149]	OPEN [87]	$n-1$ [23, 110]	TIGHT [87, 90]
Block	$\frac{n}{3} + 1$ [47] <b>[Thm 5.2]</b>	TIGHT [47] <b>[Prop 5.1]</b>	$n-1$ [23, 110]	TIGHT [87]



Table 2.5: Known bounds for ID-numbers of some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  may hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

### 2.5.1.4 Known bounds on ITD-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$\lceil \log(n+1) \rceil$ [51] <b>[Thm 3.5]</b>	TIGHT [51] <b>[Thm 3.5]</b>	$n-1$ (expect for $P_3$ ) [101]	TIGHT [101]
Paths	$\lceil \frac{3n}{5} \rceil$ ; $n$ odd $1 + \lceil \frac{3n}{5} \rceil$ ; $n$ even [123]	TIGHT [123]	$\lceil \frac{3n}{5} \rceil$ ; $n$ odd $1 + \lceil \frac{3n}{5} \rceil$ ; $n$ even [123]	TIGHT [123]
Trees	$\frac{3(n+1)}{7}$ [123]	TIGHT [123]	$n-1$ [123]	TIGHT [123]
Twin-free trees	$\frac{3(n+1)}{7}$ [123]	TIGHT [123]	$\frac{3n}{4}$ [101]	TIGHT [101]

Table 2.6: Known bounds for ITD-numbers of some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  may hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

### 2.5.1.5 Known bounds on OTD-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$\lceil \log(n+1) \rceil$ [190, 133]	TIGHT [190]	$n$ [89]	TIGHT [89]
Paths	$\lceil \frac{2n}{3} \rceil$ [24]	TIGHT [24]	$\lceil \frac{2n}{3} \rceil$ [24]	TIGHT [24]
Trees	$\lceil \frac{n}{2} \rceil + 1$ [190]	TIGHT [191]	$n-1$ [190]	TIGHT [191]
Cubic	$\frac{n}{2}$ [190]	OPEN	$\frac{3n}{4}$ [130]	TIGHT [130]
$r$ -regular	$\frac{2n}{r+1}$ [190]	OPEN	$n-1$ [89]	TIGHT [47]
Interval	$\sqrt{2n + \frac{1}{4}} - \frac{1}{2}$ [102]	TIGHT [102]	$n-1$ [89]	OPEN
Unit interval	$\frac{n+1}{2}$ [102]	TIGHT [102]	$n-1$ [89]	OPEN

Permutation	$\sqrt{n+2}$ [102]	TIGHT [102]	$n-1$ [89]	OPEN
Bipartite permutation	$\frac{n-2}{2}$ [102]	OPEN	$n-1$ [89]	OPEN
Cobipartite	$\lceil \log(n+1) \rceil$ [190, 133]	TIGHT <b>[Prop 3.5]</b>	$n-1$ [89]	OPEN
Split	$\lceil \log(n+1) \rceil$ [190, 133]	TIGHT <b>[Prop 3.5]</b>	$n-1$ [89]	OPEN
Chordal	$\lceil \log(n+1) \rceil$ [190, 133]	TIGHT <b>[Prop 3.5]</b>	$n-1$ [89]	OPEN
Block	$\frac{n+2}{3}$ [47] <b>[Thm 6.2]</b>	TIGHT [47] <b>[Prop 6.2]</b>	$n-1$ [47] <b>[Thm 6.1]</b>	TIGHT [47] <b>[Prop 6.1]</b>
Cycles	$\lceil \frac{2n}{3} \rceil$ ; $n$ odd $2 \lceil \frac{n}{3} \rceil$ ; $n$ even [26] <b>[Thm 6.4]</b>	TIGHT [26] <b>[Thm 6.4]</b>	$\lceil \frac{2n}{3} \rceil$ ; $n$ odd $2 \lceil \frac{n}{3} \rceil$ ; $n$ even [26] <b>[Thm 6.4]</b>	TIGHT [26] <b>[Thm 6.4]</b>

Table 2.7: Known bounds for OTD-numbers of some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  may hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

### 2.5.1.6 Known bounds on OD-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$\lceil \log n \rceil$ [51, 52] <b>[Thm 3.5]</b>	TIGHT [51] <b>[Prop 3.4]</b>	$n-1$ [52] <b>[Thm 3.4]</b>	TIGHT [52] <b>[Cor 6.2]</b>
Bipartite	$\lceil \log n \rceil$ [51, 52] <b>[Thm 3.5]</b>	OPEN	$n-1$ [52] <b>[Thm 3.4]</b>	TIGHT [52] <b>[Cor 6.2]</b>
Cobipartite	$\lceil \log n \rceil$ [51, 52] <b>[Thm 3.5]</b>	TIGHT [51] <b>[Prop 3.4]</b>	$n-1$ [52] <b>[Thm 3.4]</b>	OPEN
Split	$\lceil \log n \rceil$ [51, 52] <b>[Thm 3.5]</b>	TIGHT [51] <b>[Prop 3.4]</b>	$n-1$ [52] <b>[Thm 3.4]</b>	OPEN
Chordal	$\lceil \log n \rceil$ [51, 52] <b>[Thm 3.5]</b>	TIGHT [51] <b>[Prop 3.4]</b>	$n-1$ [52] <b>[Thm 3.4]</b>	OPEN

Table 2.8: Known bounds for OD-numbers of some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  may hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

### 2.5.1.7 Known bounds on FD-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$1 + \lfloor \log n \rfloor$ [51] [Thm 3.5]	TIGHT [51] [Thm 3.5]	$n$ [53]	TIGHT [53] [Cor 7.1]
Bipartite	$1 + \lfloor \log n \rfloor$ [51] [Thm 3.5]	OPEN	$n$ [53]	TIGHT [53] [Cor 7.1]
Paths	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]
Cycles	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]

Table 2.9: Known bounds for FD-numbers of some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  may hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

### 2.5.1.8 Known bounds on FTD-numbers

Graph family	Lower bound	Tightness	Upper bound	Tightness
General	$1 + \lfloor \log(n+1) \rfloor$ [51] [Thm 3.5]	TIGHT [51] [Thm 3.5]	$n$ [53]	TIGHT [53] [Cor 7.1]
Bipartite	$1 + \lfloor \log(n+1) \rfloor$ [51] [Thm 3.5]	OPEN	$n$ [53]	TIGHT [53] [Cor 7.1]
Paths	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]
Cycles	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]	$1 + \lfloor \frac{2n}{3} \rfloor$ ; $n$ odd $2 \lfloor \frac{n}{3} \rfloor$ ; $n$ even [54] [Thm 7.2]	TIGHT [54] [Thm 7.2]

Table 2.10: Known bounds for FTD-numbers of some common graph families. Here  $n$  denotes the order of a graph and the general bounds in terms of  $n$  may hold for  $n \geq n_0$  for some small  $n_0$ . The references marked in red are those that have been presented in this thesis.

## 2.5.2 Computational complexity of X-Code

We now concentrate on a literature survey portraying the algorithmic aspects of the eight identification problems. For any  $X \in \text{CODES}$ , since we are interested in finding the  $X$ -number of a given  $X$ -admissible graph, the question of finding such a minimum code is also of interest from an algorithmic and computational point of view.  $X\text{-CODE}$  for all  $X \in \text{CODES}$  is, in general, NP-complete. However, for certain graph families, these problems have been shown to be polynomial-time solvable in its input size. In Tables 2.11 – 2.18, we provide a survey of which complexity class  $X\text{-CODE}$  belongs to when the input graph is from a specific graph class.

### 2.5.2.1 Known complexity of LD-Code

Graph family	Complexity class
General	NP-complete [68, 58]
Bipartite	NP-complete [58]
Interval of diameter 2	NP-complete [103]
Permutation of diameter 2	NP-complete [103]
Subcubic planar bipartite	NP-complete [103]
Series-parallel	NP-complete [68]
Trees	P [185, 13]
Block	P [10]
Graphs of bounded clique-width	P [185, 13, 74]

Table 2.11: Table showing the complexity classes to which LD-CODE belongs when the input graph is from a specific graph class.

### 2.5.2.2 Known complexity of LTD-Code

Graph family	Complexity class
General	NP-complete [162]
Chordal bipartite	NP-complete [177]
Split	NP-complete [177]
Planar bipartite of maximum degree 3	NP-complete [177]
Planar bipartite of maximum degree 4 and girth 3	NP-complete [177]
Planar bipartite of maximum degree 4 and large girth	NP-complete [177]
Circle graphs	NP-complete [176]
Undirected path graphs	NP-complete [176]
Graphs of separability at most 2	NP-complete [176]
Trees	P [176]

Table 2.12: Table showing the complexity classes to which LTD-CODE belongs when the input graph is from a specific graph class.

### 2.5.2.3 Known complexity of ID-Code

Graph family	Complexity class
General	NP-complete [58]
Chordal bipartite	NP-complete [87]
Subcubic planar	NP-complete [14]
Planar with maximum degree 4 and large girth	NP-complete [13]

planar bipartite with maximum degree 4	NP-complete [87]
Line	NP-complete [87]
Planar bipartite unit disk	NP-complete [167]
Split	NP-complete [87]
Undirected path	NP-complete [87]
Interval of diameter 2	NP-complete [103]
Permutation of diameter 2	NP-complete [103]
Cobipartite	NP-complete [87]
Block	P [10]
Bounded treewidth/cliquewidth	P [163]
Line of bounded treewidth	P [87]

Table 2.13: Table showing the complexity classes to which ID-CODE belongs when the input graph is from a specific graph class.

#### 2.5.2.4 Known complexity of ITD-Code

Graph family	Complexity class
General	NP-complete [173]
Chordal	NP-complete [173]
Chordal bipartite	NP-complete [173]
Star-convex bipartite	NP-complete [173]
Planar	NP-complete [173]

Table 2.14: Table showing the complexity classes to which ITD-CODE belongs to when the input graph is from a specific graph class.

#### 2.5.2.5 Known complexity of OD-Code

Graph family	Complexity class
General	NP-complete [52] [ <b>Thm 8.1</b> ]
Bipartite of maximum degree 5	NP-complete [52] [ <b>Thm 8.1</b> ]

Table 2.15: Table showing the complexity classes to which OD-CODE belongs to when the input graph is from a specific graph class. The references marked in red are those that have been presented in this thesis.

### 2.5.2.6 Known complexity of OTD-Code

Graph family	Complexity class
General	NP-complete [190]
Bipartite with maximum degree 4	NP-complete [172]
Planar with maximum degree 4	NP-complete [172]
Split	NP-complete [172]
Interval of diameter 2	NP-complete [103]
Permutation of diameter 2	NP-complete [103]
Perfect elimination bipartite	NP-complete [171]
Bounded treewidth	P [172]
Paths	P [190]
Trees	P [172]
Block	P [10]
Cographs	P [103]
Chain	P [171]
Cycles	P [26] <b>[Thm 6.4]</b>

Table 2.16: Table showing the complexity classes to which OTD-CODE belongs to when the input graph is from a specific graph class. The references marked in red are those that have been presented in this thesis.

### 2.5.2.7 Known complexity of FD-Code

Graph family	Complexity class
General	NP-complete [53] <b>[Thm 8.4]</b>
Paths	P [54] <b>[Thm 7.2]</b>
Cycles	P [54] <b>[Thm 7.2]</b>

Table 2.17: Table showing the complexity classes to which FD-CODE belongs to when the input graph is from a specific graph class. The references marked in red are those that have been presented in this thesis.

### 2.5.2.8 Known complexity of FTD-Code

Graph family	Complexity class
General	NP-complete [53] <b>[Thm 8.5]</b>
Paths	P [54] <b>[Thm 7.2]</b>
Cycles	P [54] <b>[Thm 7.2]</b>

---

Table 2.18: Table showing the complexity classes to which FTD-CODE belongs to when the input graph is from a specific graph class. The references marked in red are those that have been presented in this thesis.

# Chapter 3

## A unified study of codes

In this chapter, we provide an overview of the X-codes, for all  $X \in \text{CODES}$ , and especially how one code number relates to another. In Section 3.1, we reformulate the problems of finding X-codes — and especially the minimum X-codes — in terms of covering problems of hypergraphs. Then in Section 3.2, drawing from the hypergraph reformulation of the problems, we show how the incidence matrices of these hypergraphs can be used to solve integer linear programs to also tackle the problem of finding minimum X-codes of graphs. This method described in Section 3.2 accounts for a polyhedral approach to minimizing the solution size of the problems.

Finally, in Section 3.4, we also provide some general upper and lower bounds on the code numbers of graphs. Even though many of these general bounds have already been addressed in the literature of identification problems of graphs, these questions of general bounds still need to be addressed for some of the recently introduced codes (like the OD-, FD- and FTD-codes). We do that in this section and thus, along with the bounds already established in the literature, we provide a holistic picture of all the eight codes in terms of these general bounds.

Several results in this chapter have also appeared in [51], [52], [53] and [54].

### 3.1 Hypergraph representation of codes

A very well-studied generalization of a graph is that of a *hypergraph*  $\mathcal{H} = (V(\mathcal{H}), \mathcal{E}(\mathcal{H}))$ , also called a *set system*. Like graphs, a hypergraph is also a 2-tuple, with  $V(\mathcal{H})$  being a set called the *vertex set* of  $\mathcal{H}$  and  $\mathcal{E}(\mathcal{H})$  being a set of subsets of  $V(\mathcal{H})$  called the *hyperedge set* of  $\mathcal{H}$ . See Figure 3.1 for an example of a hypergraph. As in the case of graphs, the elements of the vertex set of  $\mathcal{H}$  are also called *vertices* of  $\mathcal{H}$  and the elements of the hyperedge set of  $\mathcal{H}$  are called *hyperedges* of  $\mathcal{H}$ . A vertex subset  $S$  of  $\mathcal{H}$  is called a *transversal* or a *cover* of  $\mathcal{H}$  if  $E \cap S \neq \emptyset$  for all  $E \in \mathcal{E}(\mathcal{H})$ . A cover of a hypergraph  $\mathcal{H}$  has also been referred to in the literature as a *hitting set* or a *vertex cover* of  $\mathcal{H}$ . The minimum of the cardinalities of all covers of  $\mathcal{H}$  is called the *covering number* of  $\mathcal{H}$  and is denoted by  $\tau(\mathcal{H})$ .

Let  $G$  be a graph and let  $B \in \text{NBD-TYPE}$ . Moreover, let  $N_B(G) = \{N_B(G; v) : v \in V(G)\}$  be called the *B-neighborhood set* of  $G$ . We now define  $\mathcal{H}_{\text{dom-B}}(G)$  to be the hypergraph such that  $V(\mathcal{H}_{\text{dom-B}}(G)) = V(G)$  and  $\mathcal{E}(\mathcal{H}_{\text{dom-B}}(G)) = N_B(G)$ . Then, by Remark 2.4, the following fact is immediate.

**Remark 3.1.** *Let  $G$  be a graph and let  $B \in \text{NBD-TYPE}$ . Then, a vertex subset  $S$  of  $G$  is a B-dominating set of  $G$  if and only if  $S$  is a cover of the hypergraph  $\mathcal{H}_{\text{dom-B}}(G)$ .*

Let  $G$  be a graph and let  $A \in \text{SEP-TYPE}$ . Moreover, let  $\Delta_A(G) = \{\Delta_A(G; u, v) : u, v \in V(G) \text{ and } u \neq v\}$  be called the *A-separator set* of  $G$ . We now define  $\mathcal{H}_{\text{sep-A}}(G)$  to be the hypergraph such that  $V(\mathcal{H}_{\text{sep-A}}(G)) = V(G)$  and  $\mathcal{E}(\mathcal{H}_{\text{sep-A}}(G)) = \Delta_A(G)$ . Then, by the definition of an A-separating set, we have the following remark.



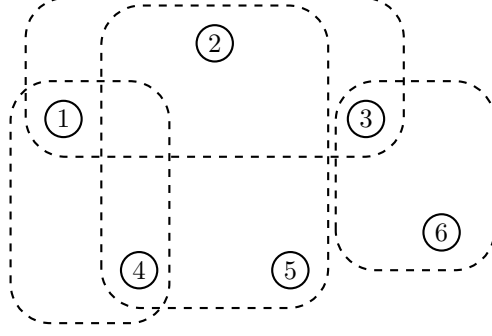


Figure 3.1: Example of a hypergraph  $\mathcal{H}$  with vertex set  $V(\mathcal{H}) = \{1, 2, 3, 4, 5, 6\}$  and hyperedge set  $\mathcal{E}(\mathcal{H}) = \{\{1, 2, 3\}, \{2, 4, 5\}, \{1, 4\}, \{3, 6\}\}$ . A dashed rectangle depicts the hyperedge which contains the vertices inside the rectangle.

**Remark 3.2.** Let  $G$  be a graph and let  $A \in \text{SEP-TYPE}$ . Then, a vertex subset  $S$  of  $G$  is an  $A$ -separating set of  $G$  if and only if  $S$  is a cover of the hypergraph  $\mathcal{H}_{\text{sep-A}}(G)$ .

Let  $\mathcal{H}_1$  and  $\mathcal{H}_2$  be any two hypergraphs. Then, let us define  $\mathcal{H}_1 \oplus \mathcal{H}_2$  to be the hypergraph with its vertex set  $V(\mathcal{H}_1) \cup V(\mathcal{H}_2)$  and hyperedge set  $\mathcal{E}(\mathcal{H}_1) \cup \mathcal{E}(\mathcal{H}_2)$ . For  $X \in \text{CODES}$ ,  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$  and for a graph  $G$ , we now look at the hypergraph  $\mathcal{H}_X(G) = \mathcal{H}_{\text{sep-A}}(G) \oplus \mathcal{H}_{\text{dom-B}}(G)$ . Then, notice that

$$V(\mathcal{H}_X(G)) = V(G) \quad \text{and} \quad E(\mathcal{H}_X(G)) = \Delta_A(G) \cup N_B(G).$$

We call the hypergraph  $\mathcal{H}_X(G)$  to be the  $X$ -hypergraph of  $G$ .

**Remark 3.3.** Let  $G$  be a graph and let  $X \in \text{CODES}$ ,  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$ . Then, a vertex subset  $S$  of a graph  $G$  is an  $X$ -code of  $G$  if and only if  $S$  is a cover of the hypergraph  $\mathcal{H}_X(G)$ . In particular, we have  $\gamma^X(G) = \tau(\mathcal{H}_X(G))$ .

Let  $A \in \text{SEP-TYPE}$ ,  $B \in \text{NBD-TYPE}$  and  $X \in \text{CODES}$  such that  $X \equiv (A, B)$ . Let  $E, E' \in \mathcal{H}_X(G)$  such that  $E \subseteq E'$ . Then, the hyperedge  $E'$  is called a *redundant hyperedge*. So, from the point of view of finding a cover of  $\mathcal{H}_X(G)$  (to find an  $X$ -code of  $G$  using Remark 3.3), any vertex subset  $S$  of  $\mathcal{H}_X(G)$  such that  $E \cap S \neq \emptyset$  automatically implies that  $E' \cap S \neq \emptyset$ . As a result, the hypergraphs  $\mathcal{H}_X(G)$  and  $\mathcal{H}_X(G) - E'$  have the same set of covers. Therefore, it is natural to work with and find the covers of the  $X$ -hypergraph  $\mathcal{H}_X(G)$  minus all the redundant hyperedges of  $\mathcal{H}_X(G)$ . Hence, as pointed out in [8, 11], to find an  $X$ -code of a graph  $G$ , we consider the  $X$ -clutter  $\mathcal{C}_X(G)$  of  $G$  obtained from  $\mathcal{H}_X(G)$  by removing all its redundant hyperedges. Since the set of covers of  $\mathcal{H}_X(G)$  and  $\mathcal{C}_X(G)$  are the same, we note that  $\tau(\mathcal{H}_X(G)) = \tau(\mathcal{C}_X(G))$  as well. As a result, we may rewrite Remark 3.3 as the following.

**Remark 3.4.** Let  $G$  be a graph and let  $X \in \text{CODES}$ ,  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$ . Then, a vertex subset  $S$  of a graph  $G$  is an  $X$ -code of  $G$  if and only if  $S$  is a cover of the  $X$ -clutter  $\mathcal{C}_X(G)$ . In particular, we have  $\gamma^X(G) = \tau(\mathcal{C}_X(G))$ .

Moreover, a special interest lies in hyperedges of  $\mathcal{C}_X(G)$  consisting of a single vertex — the forced vertices of  $G$  — as each forced vertex needs to belong to every  $X$ -code of  $G$ . For any  $X$ -code, let us denote the set of forced vertices of  $G$  by  $\mathcal{F}_X^1(G)$  which we characterize in the following manner.

**Lemma 3.1.** Let  $X \in \text{CODES}$ ,  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$ . Then, for an  $X$ -admissible graph  $G$ , we have

$$\begin{aligned} \mathcal{F}_X^1(G) = & \{w \in V(G) : \Delta_A(G; u, v) = \{w\} \text{ for some distinct } u, v \in V(G)\} \\ & \bigcup \{v \in V(G) : N_B(G; u) = \{v\} \text{ for some } u \in V(G)\}. \end{aligned}$$

As a result, we can express the X-clutter of a graph  $G$  by  $\mathcal{C}_X(G) = (V(G), \mathcal{F}_X^1(G) \cup \mathcal{F}_X^2(G))$ , where  $\mathcal{F}_X^2(G)$  is the set of all hyperedges of  $\mathcal{C}_X(G)$  of order at least 2. For illustration, we take  $X = \text{OD}$  and construct  $\mathcal{H}_{\text{OD}}(P_4)$  and  $\mathcal{C}_{\text{OD}}(P_4)$ . The OD-hypergraph  $\mathcal{H}_{\text{OD}}(P_4)$  contains the following hyperedges.

$$\begin{aligned} N[1] &= \{1, 2\} & N(1)\Delta N(2) &= \{1, 2, 3\} & N(1)\Delta N(3) &= \{4\} \\ N[2] &= \{1, 2, 3\} & N(2)\Delta N(3) &= \{1, 2, 3, 4\} & N(1)\Delta N(4) &= \{2, 3\} \\ N[3] &= \{2, 3, 4\} & N(3)\Delta N(4) &= \{2, 3, 4\} & N(2)\Delta N(4) &= \{1\} \\ N[4] &= \{3, 4\} \end{aligned}$$

Then, we can check that the OD-clutter  $\mathcal{C}_{\text{OD}}(P_4)$  only contains the symmetric differences of open neighborhoods of non-adjacent vertices, namely, the sets  $\{1\}, \{2, 3\}, \{4\}$ . Moreover, we have  $\mathcal{F}_{\text{OD}}^1(P_4) = \{\{1\}, \{4\}\}$  and  $\mathcal{F}_{\text{OD}}^2(P_4) = \{\{2, 3\}\}$ .

## 3.2 Polyhedra associated with codes

Let  $X \in \text{CODES}$  and let  $G$  be an  $X$ -admissible graph on  $n$  vertices. Let  $V(G) = \{v_1, v_2, \dots, v_n\}$  and let  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  be a column vector with each entry being a variable assuming values in  $\{0, 1\}$ . Moreover, for any vertex  $v$  of  $G$ , let  $x_v = x_i$  if  $v = v_i$ . Furthermore, let  $\mathbf{1} = (1, 1, \dots, 1)$  be a column vector with  $n$  entries with each entry being 1. Let  $m$  be the number of hyperedges of the  $X$ -clutter  $\mathcal{C}_X(G)$  and let  $\mathcal{E}(\mathcal{C}_X(G)) = \{E_1, E_2, \dots, E_m\}$ . Then, let

$$M_X(G) = (\delta_{ij})_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}}$$

be the incidence matrix of the  $X$ -clutter  $\mathcal{C}_X(G)$ , that is,  $\delta_{ij} = 1$  if the vertex  $v_j \in E_i$  and  $\delta_{ij} = 0$  otherwise. Following the approach proposed in [5, 8, 11], let us consider the following integer linear program.

$$\begin{aligned} \min \mathbf{1}^T \mathbf{x} \\ M_X(G) \mathbf{x} &\geq \mathbf{1} \\ \mathbf{x} &\in \{0, 1\}^n \end{aligned}$$

For any matrix  $M$  with  $n$  columns and having entries in  $\{0, 1\}$ , the associated covering polyhedron is  $P(M) = \text{conv}\{\mathbf{x} \in \mathbf{Z}_+^n : M\mathbf{x} \geq \mathbf{1}\}$ . Accordingly, the  $X$ -polyhedron of  $G = (V, E)$  is defined by

$$P_X(G) = P(M_X(G)) = \text{conv}\{\mathbf{x} \in \mathbf{Z}_+^n : M_X(G) \mathbf{x} \geq \mathbf{1}\}.$$

Based on results from [16] on general covering polyhedra, we have the following theorem.

**Theorem 3.1** ([8, 11, 52]). *Let  $X \in \text{CODES}$  and let  $G$  be an  $X$ -admissible graph. Then,  $P_X(G)$  has*

- (a) *the equation  $x_v = 1$  for all forced vertices of  $G$ ;*
- (b) *a nonnegativity constraint  $x_v \geq 0$  for all vertices  $v$  of  $G$  with  $\{v\} \notin \mathcal{F}_X^1(G)$ ; and*
- (c)  *$\sum_{v \in E} x_v \geq 1$  for all hyperedges  $E$  of  $\mathcal{C}_X(G)$  with  $E \in \mathcal{F}_X^2(G)$ .*

For any covering polyhedron  $P(M)$  associated with a 0/1-matrix  $M$  with  $n$  columns,  $Q(M) = \{\mathbf{x} \in \mathbf{R}_+^n : M\mathbf{x} \geq \mathbf{1}\}$  is its linear relaxation. We have  $P(M) \subseteq Q(M)$  in general and further constraints have to be added to the system  $M\mathbf{x} \geq \mathbf{1}$  in order to describe  $P(M)$  using real variables instead of integral ones.

The hypergraph  $\mathcal{R}_n^q = (V, \mathcal{E})$  with  $V = \{1, \dots, n\}$  and  $\mathcal{E}$  containing all  $q$ -element subsets of  $V$  for  $2 \leq q < n$  is called *complete  $q$ -rose of order  $n$* . In [8] it was proved that the covering polyhedron of  $\mathcal{R}_n^q$  is given by the nonnegativity constraints and

$$x(V') = \sum_{v \in V'} x_v \geq |V'| - q + 1 \tag{3.1}$$

for all subsets  $V' \subseteq \{1, \dots, n\}$  with  $|V'| \in \{q, \dots, n\}$ . Moreover, we have  $\tau(\mathcal{R}_n^q) = n - q + 1$ . Note that for  $q = 2$ ,  $\mathcal{R}_n^q$  is in fact the complete graph  $K_n$ .

### 3.3 Comparisons among codes

In this section, we compare the code-numbers of all the various codes in CODES. To that end, by the word “compare”, we precisely mean the following question.

**Question 3.1.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ - and an  $X'$ -admissible graph. Does there exist integers  $\alpha$  and  $\beta$  such that  $\gamma^{X'}(G) \leq \alpha\gamma^X(G) + \beta$ ?*

Some attempts to compare code numbers of graphs have already been made in the literature. For example, in [116], the authors have compared the LD-numbers and ID-numbers. In [194], Sewell has compared the LD-, ID- and the OTD-numbers and have established bounds for one in terms of the other. However, most of the study available with regards to such comparisons focus on a small handful of codes. In this section, we try to adopt a more general and holistic approach to answering Question 3.1 and establish such comparable relations between an  $X$ -number and an  $X'$ -number, for each pair of  $X, X' \in \text{CODES}$ .

To investigate the above question, we invoke the hypergraph representation of  $X$ -codes on an  $X$ -admissible graph  $G$  and look at the hyperedge sets of  $\mathcal{H}_X(G)$  in more detail. Part of the work in this section has also appeared in [53]

Let  $G$  be a graph and let  $A \in \text{SEP-TYPE}$ . Then, define

$$\begin{aligned}\Delta_A^1(G) &= \{\Delta_A(G; u, v) \in \Delta_A(G) : u, v \in V(G) \text{ and } d_G(u, v) = 1\}; \text{ and} \\ \Delta_A^{2+}(G) &= \{\Delta_A(G; u, v) \in \Delta_A(G) : u, v \in V(G) \text{ and } d_G(u, v) \geq 2\}.\end{aligned}$$

Then, by Observation 2.3, we have

$$\Delta_L^1(G) = \Delta_O^1(G) \quad \text{and} \quad \Delta_L^{2+}(G) = \Delta_C^{2+}(G). \quad (3.2)$$

Moreover, by Observation 2.5, we have

$$\Delta_F^1(G) = \Delta_C^1(G) \quad \text{and} \quad \Delta_F^{2+}(G) = \Delta_O^{2+}(G). \quad (3.3)$$

In other words, the hypergraphs  $\mathcal{H}_{\text{sep-L}}(G)$  and  $\mathcal{H}_{\text{sep-F}}(G)$  have two kinds of  $(u, v)$ -separators as hyperedges depending on whether two vertices  $u, v \in V(G)$  are adjacent in  $G$  or not. However, this is not the case with  $\mathcal{H}_{\text{sep-C}}(G)$  and  $\mathcal{H}_{\text{sep-O}}(G)$ . In what follows, we shall have to address these two categories of hyperedges separately. Hence, for any  $A \in \text{SEP-TYPE}$ , let us call a separator in  $\Delta_A^1(G)$  to be a *dist-1  $A$ -separator* and one in  $\Delta_A^{2+}(G)$  to be a *dist-2+  $A$ -separator*. As before, whenever the separation type  $A$  is clear from the context, we may also refer to them as a *dist-1 separator* and a *dist-2+ separator*, respectively. Moreover, for any  $X \in \text{CODES}$ ,  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$ , notice that  $\mathcal{H}_{\text{sep-A}}(G) = \mathcal{H}_{\text{sep-A}}^1(G) \oplus \mathcal{H}_{\text{sep-A}}^{2+}(G)$ , where  $\mathcal{H}_{\text{sep-A}}^1(G) = (V(G), \Delta_A^1(G))$  and  $\mathcal{H}_{\text{sep-A}}^{2+}(G) = (V(G), \Delta_A^{2+}(G))$ . Therefore, wherever necessary, we shall further expand the  $X$ -hypergraph of  $G$  to  $\mathcal{H}_X(G) = \mathcal{H}_{\text{sep-A}}^1(G) \oplus \mathcal{H}_{\text{sep-A}}^{2+}(G) \oplus \mathcal{H}_{\text{dom-B}}(G)$ .

We now begin to answer Question 3.1 starting with some general results on relations between hypergraphs sharing a common vertex set.

Let  $V$  be a fixed set and let  $\mathcal{H}^*(V)$  be the set of all hypergraphs  $\mathcal{H} = (V(\mathcal{H}), \mathcal{E}(\mathcal{H}))$  such that  $V(\mathcal{H}) = V$ . We define a relation  $\prec$  on  $\mathcal{H}^*(V)$  by  $\mathcal{H} \prec \mathcal{H}'$  if, for every hyperedge  $E'$  of  $\mathcal{H}'$ , there exists a hyperedge  $E$  of  $\mathcal{H}$  such that  $E \subseteq E'$ . We can show the following results, also laid out by Chakraborty (the author) and Wagler in [53].

**Lemma 3.2.** *If  $\mathcal{H}, \mathcal{H}' \in \mathcal{H}^*(V)$  with  $\mathcal{H} \prec \mathcal{H}'$ , then any cover of  $\mathcal{H}$  is also a cover of  $\mathcal{H}'$ . In particular, we have  $\tau(\mathcal{H}') \leq \tau(\mathcal{H})$ .*

*Proof.* Let  $\mathcal{H}, \mathcal{H}' \in \mathcal{H}^*(V)$  with  $\mathcal{E}(\mathcal{H}) = \mathcal{E}$  and  $\mathcal{E}(\mathcal{H}') = \mathcal{E}'$ , say, and assume that  $\mathcal{H} \prec \mathcal{H}'$ . Moreover, let  $S \subset V$  be a cover of  $\mathcal{H}$ . Now, let  $E' \in \mathcal{E}'$  be any hyperedge. Since  $\mathcal{H} \prec \mathcal{H}'$ , there exists a hyperedge  $E$  of  $\mathcal{H}$  such that  $E \subseteq E'$ . Then  $S \cap E \neq \emptyset$ , as  $S$  is a cover of  $\mathcal{H}$ . This implies that

$S \cap E' \neq \emptyset$  as well. Hence,  $S$  intersects every hyperedge of  $\mathcal{H}'$  and thus, is also a cover of  $\mathcal{H}'$ . The second statement follows by the fact that, if  $S$  is a minimum cover of  $\mathcal{H}$ , then  $S$  is also a cover of  $\mathcal{H}'$  and hence,  $\tau(\mathcal{H}') \leq |S| = \tau(\mathcal{H})$ .  $\square$

For any graph  $G$ ,  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$ , recall that  $N_B(G) = \{N_B(G; v) : v \in V(G)\}$  and  $\Delta_A(G) = \{\Delta_A(G; u, v) : u, v \in V(G) \text{ and } u \neq v\}$ .

**Observation 3.1.** *Given a graph  $G$ , the following hold.*

- (1) *For every set  $E' \in N_C(G)$ , there exists a set  $E \in N_O(G)$  such that  $E \subset E'$ .*
- (2) *For every set  $E' \in \Delta_O^1(G)$ , there exists a set  $E \in \Delta_C^1(G)$  such that  $E \subset E'$ .*
- (3) *For every set  $E' \in \Delta_C^{2+}(G)$ , there exists a set  $E \in \Delta_O^{2+}(G)$  such that  $E \subset E'$ .*

*Proof.* (1) Let  $E' = N_C(G; v) \in N_C(G)$  for some vertex  $v \in V(G)$ . Then, the result follows by taking  $E = N_O(G; v) \in N_O(G)$ .

(2) Let  $E' = \Delta_O(G; u, v) \in \Delta_O^1(G)$  for some pair of adjacent vertices  $u, v \in V(G)$ . In this case, we take  $E = \Delta_C(G; u, v) \in \Delta_C^1(G)$  and hence, the result follows by Observation 2.2.

(3) Let  $E' = \Delta_C(G; u, v) \in \Delta_C^{2+}(G)$  for some pair of non-adjacent vertices  $u, v \in V$ . In this case, we take  $E = \Delta_O(G; u, v) \in \Delta_O^{2+}(G)$  and hence, the result follows by Observation 2.2.  $\square$

**Corollary 3.1.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ - and  $X'$ -admissible graph. Then the following inequalities are true.*

$$\gamma^{X'}(G) \leq \gamma^X(G) \text{ for } \begin{cases} X = LTD, & X' = LD; \\ X = ITD, & X' = ID; \\ X = OTD, & X' = OD; \\ X = FTD, & X' = FD. \end{cases}$$

*Proof.* Here, an  $X'$ -code and an  $X$ -code differ only in the domination type. Therefore, we have  $X' \equiv (A, C)$  and  $X \equiv (A, O)$  for some  $A \in \text{SEP-TYPE}$ . Thus, the hyperedge sets of  $\mathcal{H}_{X'}(G)$  and  $\mathcal{H}_X(G)$  differ only in that the first one contains the hyperedge subset  $\mathcal{E}(\mathcal{H}_{\text{dom-C}}(G)) = N_C(G)$  and the second one contains the hyperedge subset  $\mathcal{E}(\mathcal{H}_{\text{dom-O}}(G)) = N_O(G)$ . Therefore, using Observation 3.1(1), we have  $\mathcal{H}_X(G) \prec \mathcal{H}_{X'}(G)$ . Hence, by Lemma 3.2, we have  $\gamma^{X'}(G) = \tau(\mathcal{H}_{X'}(G)) \leq \tau(\mathcal{H}_X(G)) = \gamma^X(G)$ .  $\square$

Corollary 3.1 answers Question 3.1 for certain  $X, X' \in \text{CODES}$  with the value of  $(\alpha, \beta) = (1, 0)$ . These relations  $\gamma^{X'}(G) \leq \gamma^X(G)$  in Corollary 3.1 are depicted by the dashed arrows with labels  $(\alpha, \beta) = (1, 0)$  from the node  $X'$  to  $X$  in Figure 3.2. Lemma 3.2 and Observation 3.1 also imply the following corollary.

**Corollary 3.2.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ -admissible and  $X'$ -admissible graph. Then the following inequalities are true.*

$$\gamma^{X'}(G) \leq \gamma^X(G) \text{ for } \begin{cases} X = ID, & X' = LD; \\ X = FD, & X' = OD; \\ X = ITD, & X' = LTD; \\ X = FTD, & X' = OTD. \end{cases}$$

*Proof.* In this case, we notice that an  $X'$ -code and an  $X$ -code have the same domination type. Moreover, in the case when  $X' = LD$  and  $X = ID$ , or when  $X' = LTD$ ,  $X = ITD$ , an  $X'$ -code is a locating set and an  $X$ -code is a closed-separating set. Hence, we have  $\mathcal{E}(\mathcal{H}_{\text{sep-L}}(G)) = \Delta_O^1(G) \cup \Delta_C^{2+}$  (using Equation 3.2) and  $\mathcal{E}(\mathcal{H}_{\text{sep-C}}(G)) = \Delta_C^1(G) \cup \Delta_C^{2+}$ . Therefore, an  $X'$ -code and an  $X$ -code differ only in the dist-1 separators. Similarly, in the case that  $X' = OD$ ,  $X = FD$  or that  $X' = OTD$ ,  $X = FTD$ , an  $X'$ -code is an open-separating set and an  $X$ -code is a full-separating set. Hence, we have  $\mathcal{E}(\mathcal{H}_{\text{sep-O}}(G)) = \Delta_O^1(G) \cup \Delta_O^{2+}$  and  $\mathcal{E}(\mathcal{H}_{\text{sep-F}}(G)) = \Delta_C^1(G) \cup \Delta_O^{2+}$  (using Equation 3.3).

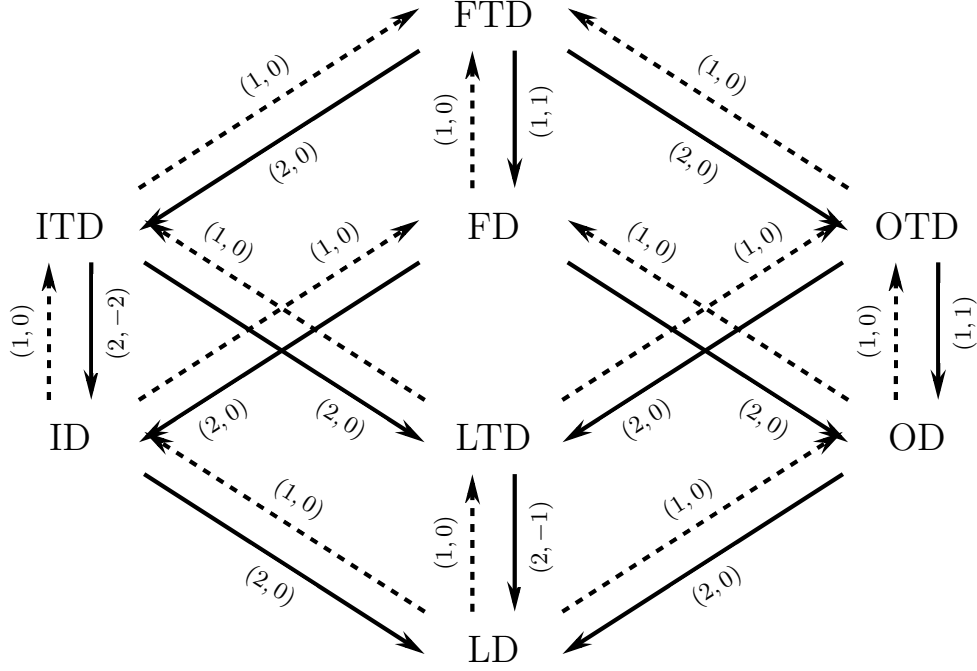


Figure 3.2: The directed multicube  $C_{\text{CODES}}$  with vertex set CODES. A dashed arc  $X' \rightarrow X$  exists if  $\gamma^{X'}(G) \leq \gamma^X(G)$  in Corollaries 3.1 to 3.3; and a solid arc  $X' \rightarrow X$  exists if  $\gamma^{X'}(G) \leq \alpha\gamma^X(G) + \beta$  in Corollaries 3.4 to 3.6 and Remark 3.6, where  $\alpha, \beta \in \mathbb{N}$ . Moreover, each arc  $X' \rightarrow X$  (dashed or solid) signifying a relation  $\gamma^{X'}(G) \leq \alpha\gamma^X(G) + \beta$  is labeled with the 2-tuple  $(\alpha, \beta)$ .

Therefore, here again, an  $X'$ -code and an  $X$ -code differ only in the dist-1 separators. This implies that, in all the four cases, the hyperedge sets of  $\mathcal{H}_{X'}(G)$  and  $\mathcal{H}_X(G)$  differ only in that the first one contains the hyperedge subset  $\Delta_0^1(G)$  and the second one contains the hyperedge subset  $\Delta_C^1(G)$ . Therefore, by Observation 3.1(2), we have  $\mathcal{H}_X(G) \prec \mathcal{H}_{X'}(G)$ . Hence, by Lemma 3.2, we have  $\gamma^{X'}(G) = \tau(\mathcal{H}_{X'}(G)) \leq \tau(\mathcal{H}_X(G)) = \gamma^X(G)$ .  $\square$

Corollary 3.2 also answers Question 3.1 for certain  $X, X' \in \text{CODES}$  with the value of  $(\alpha, \beta) = (1, 0)$ . Again, these relations  $\gamma^{X'}(G) \leq \gamma^X(G)$  in Corollary 3.2 are depicted by the dashed arrows with labels  $(\alpha, \beta) = (1, 0)$  from the node  $X'$  to  $X$  in Figure 3.2.

**Corollary 3.3.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ - and  $X'$ -admissible graph. Then the following inequalities are true.*

$$\gamma^{X'}(G) \leq \gamma^X(G) \text{ for } \begin{cases} X = OD, & X' = LD; \\ X = FD, & X' = ID; \\ X = OTD, & X' = LTD; \\ X = FTD, & X' = ITD. \end{cases}$$

*Proof.* We notice that an  $X'$ -code and an  $X$ -code have the same domination type. Moreover, in the case when  $X' = LD$  and  $X = OD$ , or when  $X' = LTD$ ,  $X = OTD$ , an  $X'$ -code is a locating set and an  $X$ -code is an open-separating set. Hence, we have  $\mathcal{E}(\mathcal{H}_{\text{sep-L}}(G)) = \Delta_0^1(G) \cup \Delta_C^{2+}$  (using Equation 3.2) and  $\mathcal{E}(\mathcal{H}_{\text{sep-O}}(G)) = \Delta_0^1(G) \cup \Delta_0^{2+}$ . Therefore, an  $X'$ -code and an  $X$ -code differ only in the dist-2+ separators. Similarly, in the case that  $X' = ID$ ,  $X = FD$  or that  $X' = ITD$ ,  $X = FTD$ , an  $X'$ -code is a closed-separating set and an  $X$ -code is a full-separating set. Hence, we have  $\mathcal{E}(\mathcal{H}_{\text{sep-C}}(G)) = \Delta_C^1(G) \cup \Delta_C^{2+}$  and  $\mathcal{E}(\mathcal{H}_{\text{sep-F}}(G)) = \Delta_C^1(G) \cup \Delta_0^{2+}$  (using Equation 3.3). Therefore, here again, an  $X'$ -code and an  $X$ -code differ only in the dist-2+ separators. This implies that, in all the four cases, the hyperedge sets of  $\mathcal{H}_{X'}(G)$  and  $\mathcal{H}_X(G)$  differ only in that the first one

contains the hyperedge subset  $\Delta_C^{2+}(G)$  and the second one contains the hyperedge subset  $\Delta_O^{2+}(G)$ . Therefore, by Observation 3.1(2), we have  $\mathcal{H}_X(G) \prec \mathcal{H}_{X'}(G)$ . Hence, by Lemma 3.2, we have  $\gamma^{X'}(G) = \tau(\mathcal{H}_{X'}(G)) \leq \tau(\mathcal{H}_X(G)) = \gamma^X(G)$ .  $\square$

Once again, Corollary 3.3 answers Question 3.1 for certain  $X, X' \in \text{CODES}$  with the value of  $(\alpha, \beta) = (1, 0)$ . These relations  $\gamma^{X'}(G) \leq \gamma^X(G)$  in Corollary 3.3 are depicted by the dashed arrows with labels  $(\alpha, \beta) = (1, 0)$  from the node  $X'$  to  $X$  in Figure 3.2.

Figure 3.2 represents a directed multicube which we shall from now on denote by  $\vec{C}_{\text{CODES}}$ . Notice that the vertex set of  $\vec{C}_{\text{CODES}}$  is the set  $\text{CODES}$ . So far, we have explained the presence of the dashed arcs along with their labels in  $\vec{C}_{\text{CODES}}$  to correspond to each of the relations laid out in Corollaries 3.1, 3.2 and 3.3. We can now ask whether all the dashed arcs in  $\vec{C}_{\text{CODES}}$  in Figure 3.2 can be reversed with some other labels. More specifically, we can ask the following question.

**Question 3.2.** *For any pair  $X, X' \in \text{CODES}$  for which  $\gamma^{X'}(G) \leq \gamma^X(G)$  in Corollaries 3.1, 3.2 and 3.3, does there exist integers  $\alpha$  and  $\beta$  such that  $\gamma^X(G) \leq \alpha\gamma^{X'}(G) + \beta$ ?*

To answer Question 3.2, we next invoke a generalized version of Bondy's Theorem [29] that has been formulated by Chakraborty (the author) et al. in [42]. Let  $G$  be a graph and let  $Z$  and  $S$  be two (not necessarily disjoint) vertex subsets of  $G$ . Moreover, let  $A \in \{C, O\}$ . Then,  $S$  induces a partition of  $Z$  by the equivalence relation  $(-, -)_S$  on  $Z$  such that  $(u, v)_S$  if  $\Delta_A(G; u, v) \cap S = \emptyset$  for any  $u, v \in Z$ . Let this partition be denoted as  $\mathcal{P}_A(S; Z)$ .

**Remark 3.5.** *Let  $G$  be a graph, let  $Z \subseteq V(G)$  and let  $A \in \{C, O\}$ . Then there exists another set  $S \subseteq V(G)$  such that each part of the partition  $\mathcal{P}_A(S; Z)$  is a singleton set if and only if  $\Delta_A(G; u, v) \cap S \neq \emptyset$  for all distinct  $u, v \in Z$ .*

Remark 3.5 therefore shows that a graph  $G$  has an  $A$ -separating set  $S$  if and only if each part of  $\mathcal{P}_A(S; V(G))$  is a singleton. For a subset  $Z \subseteq V(G)$ , if there exists another set  $S \subseteq V(G)$  such that each part of the partition  $\mathcal{P}_A(S; Z)$  is a singleton set, then the set  $S$  is called an  $A$ -separating set of  $G[Z]$  in  $G$ . Conversely, if there exists an  $A$ -separating set of  $G[Z]$  in  $G$ , then  $G[Z]$  is called  $A$ -separable in  $G$ . Notice that, when  $Z = V(G)$ , this is consistent with our previous definitions of an  $A$ -separating set and an  $A$ -separable graph.

The following lemma is based on a result by Chakraborty (the author) et al. in [42]. Even though the result was proven in the context of identifying codes and hence, closed separation, the proof has been adapted here to prove similar results even in the context of open separation. Similar approaches have also been adopted in [116, 194].

**Lemma 3.3.** *Let  $G$  be a graph, let  $A \in \{C, O\}$  and let  $Z \subseteq V(G)$  such that  $G[Z]$  is  $A$ -separable in  $G$ . Then, there exists an  $A$ -separating set  $S$  of  $G[Z]$  in  $G$  such that  $|S| \leq |Z| - 1$ .*

*Proof.* If  $|Z| = 1$ , then the result follows trivially. Therefore, we assume that  $|Z| \geq 2$ . We now inductively construct an  $A$ -separating set  $S$  of  $Z$  such that  $|S| \leq |Z| - 1$ . To begin with, let  $u, v$  be an arbitrary pair of distinct vertices of  $Z$  and let  $y \in V(G)$  such that  $y$  separates the pair (such a vertex  $y$  exists since  $G[Z]$  is  $A$ -separable in  $G$ ). Then, we let  $S = \{y\}$ . Now,  $\mathcal{P}_A(S; Z)$  is the partition induced by  $S$  on  $Z$ . Then, for as long as there exists a part  $P$  of  $\mathcal{P}_A(S; Z)$  such that  $u', v' \in P$  for two distinct vertices  $u', v'$  of  $Z$ , the construction of  $S$  follows inductively by selecting an element  $y' \in V(G)$  such that  $y'$  separates  $u', v'$ , and letting  $y' \in S$ . At each step, since  $G[Z]$  is  $A$ -separable, such a vertex  $y'$  exists. Moreover, we notice that at each inductive step, we have  $|S| \leq |\mathcal{P}_A(S; Z)| - 1$ , since at each step, we increase the number of parts by at least 1, and the cardinality of  $S$  by exactly 1. This implies that we must have  $|S| \leq |Z| - 1$ , since  $|\mathcal{P}_A(S; Z)| \leq |Z|$ . This proves the result.  $\square$

**Lemma 3.4.** *Let  $G$  be a graph and let  $A \in \{C, O\}$  such that  $\Delta_A(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . Moreover, let  $S$  be a non-empty vertex subset of  $G$  and let  $\mathcal{P}_A(S; V(G))$  be such that, for any part  $P \in \mathcal{P}_A(S; V(G))$ , we have  $|P \cap S| \geq |P| - 1$ . Then, there exists an  $A$ -separating set  $R$  of  $G[Z]$  in  $G$  such that  $|R| \leq 2|S|$  and  $R = S \cup (\cup_{P \in \mathcal{P}_A(S; Z)} R_P)$ , where  $R_P$  is an  $A$ -separating set of  $G[P]$  in  $G$ .*

*Proof.* Let  $\mathcal{P} = \mathcal{P}_A(S; V(G))$ . Since  $\Delta_A(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ , the set  $V(G)$  is an  $A$ -separating set of  $G$ . Therefore, for each  $P \in \mathcal{P}$ , there exists an  $A$ -separating set of  $G[P]$  in  $G$ , namely the set  $V(G)$ . In other words, for each  $P \in \mathcal{P}$ , the subgraph  $G[P]$  is  $A$ -separable in  $G$ . Then, by Lemma 3.3, there exists in  $G$  an  $A$ -separating set, say  $R_P$ , of  $G[P]$  such that  $|R_P| \leq |P| - 1$ . Let  $R = S \cup (\cup_{P \in \mathcal{P}} R_P)$ . Then, we have

$$|R| \leq |S| + \sum_{P \in \mathcal{P}} |R_P| \leq |S| + \sum_{P \in \mathcal{P}} (|P| - 1) \leq |S| + \sum_{P \in \mathcal{P}} |P \cap S| = 2|S|.$$

We now show that  $R$  is a separating set of  $G$  of type  $A$ . To do so, let us assume  $u, v$  to be any two distinct vertices of  $G$ . If  $u \in P_u$  and  $v \in P_v$  for two distinct parts  $P_u, P_v \in \mathcal{P}$ , then, by definition of the partition  $\mathcal{P}$ , we have  $\Delta_A(G; u, v) \cap S \neq \emptyset$ . Let us therefore assume that  $u, v \in P$  for some part  $P \in \mathcal{P}$ . Since  $R_P$  is a  $P$ -separating set of type  $A$ , again by definition, we have  $\Delta_A(G; u, v) \cap R_P \neq \emptyset$ . This proves that  $R$  is a separating set of  $G$  of type  $A$ .  $\square$

**Lemma 3.5.** *Let  $G$  be a twin-free graph,  $A \in \{C, O\}$  and let  $S$  be an  $A$ -separating set of  $G$ . Then there exists a full-separating set  $R$  of  $G$  such that  $S \subseteq R$  and  $|R| \leq 2|S|$ .*

*Proof.* Let  $A^* \in \{C, O\}$  such that  $A^* \neq A$ . Moreover, let  $\mathcal{P} = \mathcal{P}_{A^*}(S; V(G))$ . If  $|P| = 1$  for every part  $P \in \mathcal{P}$ , then the set  $S$  is already an  $A^*$ -separating set of  $G$  and hence, is a full-separating set of  $G$ . Therefore, in this case, the result follows immediately. Let us therefore assume that there exists a part  $P \in \mathcal{P}$  such that  $|P| > 1$  and let  $u, v \in P$ . Now, if  $\{u, v\} \cap S = \emptyset$ , then we have  $N_A(G; u) \cap S = N_{A^*}(G; u) \cap S$ . This implies that  $\Delta_A(G; u, v) = \Delta_{A^*}(G; u, v)$  for all distinct  $u, v \in V(G)$ . Moreover, we have  $\Delta_A(G; u, v) \cap S \neq \emptyset$ , as  $S$  is an  $A$ -separating set of  $G$ . Therefore, we have  $\Delta_{A^*}(G; u, v) \cap S \neq \emptyset$  as well. However, this contradicts the fact that  $u$  and  $v$  belong to the same part  $P$ . This implies that  $\{u, v\} \cap S \neq \emptyset$ . In other words, at least all-but-one vertices of  $P$  must belong to  $S$ , that is,  $|P \cap S| \geq |P| - 1$ . Then, by Lemma 3.4, there exists an  $A^*$ -separating set  $R$  of  $G$  such that  $|R| \leq 2|S|$  and  $R = S \cup (\cup_{P \in \mathcal{P}_S} R_P)$ , where  $R_P$  is an  $A^*$ -separating set of  $G[P]$  in  $G$ . Therefore, the set  $R$  is a full-separating set of  $G$ .  $\square$

**Corollary 3.4.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ - and  $X'$ -admissible graph. Then the following inequalities are true.*

$$\gamma^{X'}(G) \leq 2\gamma^X(G) \text{ for } \begin{cases} X = OD, & X' = FD; \\ X = ID, & X' = FD; \\ X = OTD, & X' = FTD; \\ X = ITD, & X' = FTD. \end{cases}$$

*Proof.* Here, an  $X'$ -code and an  $X$ -code have the same domination type. Let  $X \in \{OD, ID, OTD, ITD\}$  and let  $S$  be a minimum  $X$ -code of  $G$ , that is,  $|S| = \gamma^X(G)$ . Then,  $S$  is a separating set of  $G$  of type  $A \in \{C, O\}$ . Therefore, by Lemma 3.5, there exists a full-separating set, say  $R$ , of  $G$  such that  $|R| \leq 2|S| = 2\gamma^X(G)$ . Moreover, again by Lemma 3.5, since  $S \subseteq R$ , the two sets have the same domination type. This implies the result.  $\square$

Therefore, Corollary 3.4 answers question 3.2 with  $(\alpha, \beta) = (2, 0)$ . These relations  $\gamma^{X'}(G) \leq 2\gamma^X(G)$  in Corollary 3.4 are depicted by the solid arrows with labels  $(\alpha, \beta) = (2, 0)$  from the node  $X'$  to  $X$  in Figure 3.2. The following lemma about locating sets is analogous (but in reverse) to Lemma 3.5 for full-separating sets.

**Lemma 3.6.** *Let  $A \in \{C, O\}$  and let  $G$  be a graph such that  $\Delta_A(G; u, v) \neq \emptyset$  for all distinct  $u, v \in V(G)$ . Also, let  $S$  be a locating set of  $G$ . Then there exists an  $A$ -separating set  $R$  of  $G$  such that  $S \subseteq R$  and  $|R| \leq 2|S|$ .*

*Proof.* For  $A = C$ , the result is contained in [116] by Gravier et al. where the authors show that if  $S$  is an LD-code, then there exists an ID-code  $R$  of  $G$  such that  $|R| \leq 2|S|$ . However, as a unified approach to proving the result for both the cases when  $A = C$  and  $A = O$ , we present the following proof.

Let  $A \in \{C, O\}$  and let  $\mathcal{P} = \mathcal{P}_A(S; V(G))$ . If  $|P| = 1$  for every part  $P \in \mathcal{P}$ , then the set  $S$  is already an  $A$ -separating set of  $G$ . Therefore, in this case, the result follows immediately. Let us therefore assume that there exists a part  $P \in \mathcal{P}$  be such that  $|P| > 1$  and let  $u, v \in P$ . Now, if  $\{u, v\} \cap S = \emptyset$ , then we have  $N_C(G; u) \cap S = N_O(G; u) \cap S$  and  $N_C(G; v) \cap S = N_O(G; v) \cap S$ . This implies that  $\Delta_C(G; u, v) = \Delta_O(G; u, v)$  for all distinct  $u, v \in V(G)$ . Moreover, we have  $\Delta_O(G; u, v) \cap S \neq \emptyset$ , as  $S$  is a locating set of  $G$ . Therefore, we have  $\Delta_C(G; u, v) \cap S \neq \emptyset$  as well. This implies that  $\Delta_A(G; u, v) \cap S \neq \emptyset$ . However, this contradicts the fact that  $u$  and  $v$  belong to the same part  $P$ . This implies that  $\{u, v\} \cap S \neq \emptyset$ . In other words, at least all-but-one vertices of  $P$  must belong to  $S$ , that is,  $|P \cap S| \geq |P| - 1$ . Then, by Lemma 3.4, there exists an  $A$ -separating set  $R$  of  $G$  such that  $|R| \leq 2|S|$  and that  $R = S \cup (\cup_{P \in \mathcal{P}_S} R_P)$ , where  $R_P$  is an  $A$ -separating set of  $G[P]$  in  $G$ .  $\square$

**Corollary 3.5.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ - and  $X'$ -admissible graph. Then the following inequalities are true.*

$$\gamma^{X'}(G) \leq 2\gamma^X(G) \text{ for } \begin{cases} X = LD, & X' = OD; \\ X = LD, & X' = ID; \\ X = LTD, & X' = OTD; \\ X = LTD, & X' = ITD. \end{cases}$$

*Proof.* We notice that an  $X'$ -code and an  $X$ -code have the same domination type. Let  $X \in \{LD, LTD\}$  and let  $S$  be a minimum  $X$ -code of  $G$ , that is,  $|S| = \gamma^X(G)$ . Then,  $S$  is a locating set of  $G$ . Therefore, by Lemma 3.6, there exists a separating set of type  $A$  of  $G$ , say  $R$ , such that  $|R| \leq 2|S| = 2\gamma^X(G)$ , where  $A \in \{C, O\}$ . Moreover, again by Lemma 3.6, since  $S \subseteq R$ , the two sets have the same domination type. This implies the result.  $\square$

The result for  $X' = ID$  and  $X = LD$  in Corollary 3.5 has already been shown by Moncel et al. in [116]. Again, Corollary 3.5 answers question 3.2 with  $(\alpha, \beta) = (2, 0)$ . These relations  $\gamma^{X'}(G) \leq 2\gamma^X(G)$  in Corollary 3.5 are depicted by the solid arrows with labels  $(\alpha, \beta) = (2, 0)$  from the node  $X'$  to  $X$  in Figure 3.2.

**Observation 3.2.** *Let  $G$  be an isolate-free graph and let  $S$  be a dominating set of  $G$ . Then, there exists a total-dominating set  $R$  of  $G$  such that  $|R| \leq 2|S|$ .*

*Proof.* Since  $G$  is isolate-free, for every vertex  $v$  of  $G$ , we fix a neighbor  $u_v \in N_G(v)$ . Then, the result follows by letting  $R = S \cup \{u_v : v \in V(G)\}$ .  $\square$

Observation 3.2 readily implies that  $\gamma^{\text{ITD}}(G) \leq 2\gamma^{\text{ID}}(G)$  and  $\gamma^{\text{LTD}}(G) \leq 2\gamma^{\text{LD}}(G)$  for all isolate-free and closed-twin-free graphs  $G$ . However, in [101], Foucaud and Lehtilä prove the following.

**Remark 3.6.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ - and  $X'$ -admissible graph. Then, we have*

$$\begin{aligned} \gamma^{X'}(G) &\leq 2\gamma^X(G) - 2 \text{ for } X' = ITD, X = ID; \text{ and} \\ \gamma^{X'}(G) &\leq 2\gamma^X(G) - 1 \text{ for } X' = LTD, X = LD. \end{aligned}$$

Moreover, the bounds in Remark 3.6 have also been proven to be tight by the authors in [101]. Therefore, these relations are depicted by the solid arrows with labels  $(\alpha, \beta) = (2, -2)$  and  $(\alpha, \beta) = (2, -1)$ , respectively, from the node  $X'$  to  $X$  in Figure 3.2.

The following result also appears in [52, 53].

**Corollary 3.6.** *Let  $X, X' \in \text{CODES}$  and let  $G$  be an  $X$ - and  $X'$ -admissible graph. Then the following inequalities are true.*

$$\gamma^{X'}(G) \leq \gamma^X(G) + 1 \text{ for } \begin{cases} X = OD, & X' = OTD; \\ X = FD, & X' = FTD. \end{cases}$$



*Proof.* Notice that an  $X$ -code and an  $X'$ -code differ only in the domination type. Let  $X \in \{\text{OD}, \text{FD}\}$  and let  $S$  be a minimum  $X$ -code of  $G$ , that is,  $|S| = \gamma^X(G)$ . Then  $S$  is an open-separating set of  $G$ . If  $S$  is also a total-dominating set of  $G$ , then the result follows by observing that  $S$  is also an OTD-code if  $X = \text{OD}$  of  $G$ , and that it is also an FTD-code of  $G$  if  $X = \text{FD}$ . Therefore, let us assume that there exists a vertex  $v$  of  $G$  such that  $N_G(v) \cap S = \emptyset$ . If there exists another vertex  $u$  of  $G$  with  $u \neq v$  such that  $N_G(u) \cap S = \emptyset$ , then it implies that  $\Delta_O(G; u, v) \cap S = \emptyset$  thus contradicting the fact that  $S$  is an open-separating set of  $G$ . Therefore, there exists at most one vertex  $v$  of  $G$  such that  $N_G(v) \cap S = \emptyset$ . Since  $G$  is either OTD-admissible or FTD-admissible, it implies that  $G$  is isolate-free. Thus, there exists a vertex  $w \in N_G(v)$ . Then, the set  $S \cup \{w\}$  is a total-dominating set of  $G$  and hence, the result follows.  $\square$

For  $X = \text{OD}$  and  $X = \text{FD}$ , the results in Corollary 3.6 also appear individually in [52] and [53], respectively, by Chakraborty (the author) and Wagler. Corollary 3.6 answers Question 3.2 with  $(\alpha, \beta) = (1, 1)$ . The relations  $\gamma^{X'}(G) \leq \gamma^X(G) + 1$  in Corollary 3.6 are depicted by the solid arrows with labels  $(\alpha, \beta) = (1, 1)$  from the node  $X'$  to  $X$  in Figure 3.2.

Thus, Corollaries 3.4, 3.5 and 3.6 and Remark 3.6 imply that we can answer positively to Question 3.2. This explains also the solid arcs and their labels in the directed multicube  $\vec{C}_{\text{CODES}}$  depicted in Figure 3.2.

**Lemma 3.7.** *Let  $X, X' \in \text{CODES}$  and, for  $k \in [3]$ , let  $P = X_1 X_2 \dots X_{k+1}$  be a directed path in  $\vec{C}_{\text{CODES}}$  as in Figure 3.2 such that  $X_1 = X'$  and  $X_{k+1} = X$ . Moreover, let  $(\alpha_i, \beta_i)$  be the label on arc  $X_i X_{i+1}$  of  $P$  for all  $i \in [k]$ . Also, let  $\alpha_0 = 1$ . Then, for all  $X$ -admissible and  $X'$ -admissible graphs  $G$ , we have*

$$\gamma^{X'}(G) \leq \alpha \gamma^X(G) + \beta, \text{ where } \alpha = \prod_{i=1}^k \alpha_i \quad \text{and} \quad \beta = \sum_{i=1}^k \beta_i \prod_{j=0}^{i-1} \alpha_j.$$

*Proof.* The proof is by induction on  $k$ . If  $k = 1$ , then the path  $P = X'X$  is an arc in  $\vec{C}_{\text{CODES}}$  with label  $(\alpha_1, \beta_1)$ . This implies by one of the Corollaries 3.1 – 3.6 or Remark 3.6, that  $\gamma^{X'}(G) \leq \alpha_1 \gamma^X(G) + \beta_1$  and hence, the result is true with  $\alpha = \alpha_1$  and  $\beta = \alpha_0 \beta_1$ . Hence, let us assume the result to be true for  $\ell \in [2]$ . Then, for the path  $X_1 X_2 \dots X_k$ , by the induction hypothesis, we have

$$\gamma^{X'}(G) \leq \alpha' \gamma^{X_k}(G) + \beta', \text{ where } \alpha' = \prod_{i=1}^{k-1} \alpha_i \quad \text{and} \quad \beta' = \sum_{i=1}^{k-1} \beta_i \prod_{j=0}^{i-1} \alpha_j.$$

Also, for the arc  $X_k X_{k+1}$ , since  $\gamma^{X_k}(G) \leq \alpha_k \gamma^X(G) + \beta_k$ , we have

$$\gamma^{X'}(G) \leq \alpha' \gamma^{X_k}(G) + \beta' \leq \alpha' (\alpha_k \gamma^X(G) + \beta_k) + \beta' = \alpha' \alpha_k \gamma^X(G) + \beta' + \alpha' \beta_k.$$

which implies that

$$\alpha = \alpha' \alpha_k = \prod_{i=1}^k \alpha_i \quad \text{and} \quad \beta = \beta' + \alpha' \beta_k = \sum_{i=1}^k \beta_i \prod_{j=0}^{i-1} \alpha_j.$$

$\square$

**Theorem 3.2.** *Let  $G$  be an FTD-admissible (isolate-free and twin-free) graph. Then, any pair of code-numbers on  $G$  are comparable: given any pair of  $X \in \text{CODES}$ , there exist integers  $\alpha$  and  $\beta$  such*

that  $\gamma^{X'}(G) \leq \alpha\gamma^X(G) + \beta$ . Moreover, for any  $X \in \text{CODES}$ , we have

$$(1) \quad \gamma^{\text{LD}}(G) \leq \gamma^X(G) \leq \gamma^{\text{FTD}}(G) \quad [53]; \text{ and}$$

$$(2) \quad \gamma^{\text{FTD}}(G) \leq \alpha\gamma^X(G) + \beta, \text{ where}$$

$$\alpha = \min \left\{ \prod_{i=1}^k \alpha_i : P = X_1 X_2 \dots X_k \text{ directed path in } \vec{\mathcal{C}}_{\text{CODES}} \text{ with } X_1 = \text{FTD} \text{ and } X_k = X \text{ and} \right. \\ \left. \text{label } (\alpha_i, \beta_i) \text{ on the arc } X_i X_{i+1} \right\} \quad \text{and} \quad \beta = \sum_{i=1}^k \beta_i \prod_{j=0}^{i-1} \alpha_j.$$

*Proof.* In the directed multicube  $\vec{\mathcal{C}}_{\text{CODES}}$ , the vertex LD is a source and the vertex FTD is a sink with respect to the dashed arcs; whereas, with respect to the solid arcs, the vertex FTD is a source and the vertex LD is a sink of  $\vec{\mathcal{C}}_{\text{CODES}}$ . This implies that for any two vertices  $X, X'$  of  $\vec{\mathcal{C}}_{\text{CODES}}$ , there exists a directed path from  $X'$  to  $X$ . Therefore, by Lemma 3.7, there exist integers  $\alpha$  and  $\beta$  such that  $\gamma^{X'}(G) \leq \alpha\gamma^X(G) + \beta$ .

(1) Let  $\vec{P}_B(\text{LD}, X)$  be a directed path on the dashed arcs from LD to  $X$ . Then, on this path, the label on every arc is  $(1, 0)$ . Therefore, by Lemma 3.7, we have  $\gamma^{\text{LD}}(G) \leq \alpha\gamma^X(G) + \beta$  with  $\alpha = 1$  and  $\beta = 0$ . Moreover, let  $\vec{P}_B(X, \text{FTD})$  be a directed path on the dashed arcs from  $X$  to FTD. Then again, each arc on this path has a label  $(1, 0)$ . Thus, as before, by Lemma 3.7, we have  $\gamma^X(G) \leq \alpha\gamma^{\text{FTD}}(G) + \beta$  with  $\alpha = 1$  and  $\beta = 0$ . This proves the first set of inequalities.

(2) As argued before, there exists a directed path from the vertex FTD to  $X$  in the graph  $\vec{\mathcal{C}}_{\text{CODES}}$ . Then, by Lemma 3.7, we have  $\gamma^{\text{FTD}}(G) \leq \alpha\gamma^X(G) + \beta$ , where

$$\alpha = \prod_{i=1}^k \alpha_i \quad \text{and} \quad \beta = \sum_{i=1}^k \beta_i \prod_{j=0}^{i-1} \alpha_j$$

for some  $k \in [3]$  such that  $P = X_1 X_2 \dots X_k$  is a directed path in  $\vec{\mathcal{C}}_{\text{CODES}}$  with  $X_1 = \text{FTD}$  and  $X_k = X$  and with label  $(\alpha_i, \beta_i)$  on the arc  $X_i X_{i+1}$ . Hence, the result follows from choosing the directed path  $P$  that minimizes the quantity  $\prod_{i=1}^k \alpha_i$ .  $\square$

**Corollary 3.7.** *Let  $G$  be an FTD-admissible (isolate-free and twin-free) graph. Then, we have*

$$\gamma^{\text{LD}}(G) \leq \gamma^{\text{FTD}}(G) \leq 4\gamma^{\text{LD}}(G) + 2.$$

*Proof.* The inequality  $\gamma^{\text{LD}}(G) \leq \gamma^{\text{FTD}}(G)$  follows from Theorem 3.2(1). We therefore prove the inequality  $\gamma^{\text{FTD}}(G) \leq 4\gamma^{\text{LD}}(G) + 2$ . To do so, we notice that in the directed multicube  $\vec{\mathcal{C}}_{\text{CODES}}$ , the directed path  $P = X_1 X_2 X_3 X_4$  with  $X_1 = \text{FTD}$ ,  $X_2 = \text{OTD}$ ,  $X_3 = \text{OD}$  and  $X_4 = \text{LD}$  has the labels  $(\alpha_1, \beta_1) = (2, 0)$ ,  $(\alpha_2, \beta_2) = (1, 1)$  and  $(\alpha_3, \beta_3) = (2, 0)$  on the arcs  $X_1 X_2$ ,  $X_2 X_3$  and  $X_3 X_4$ , respectively. Therefore, furnishing these labels in Lemma 3.7, we have  $\gamma^{\text{FTD}}(G) \leq \alpha\gamma^{\text{LD}}(G) + \beta$  with  $\alpha = \alpha_1 \alpha_2 \alpha_3 = 4$  and  $\beta = \beta_1 + \alpha_1 \beta_2 + \alpha_1 \alpha_2 \beta_3 = 2$ . This proves the result.  $\square$

## 3.4 General bounds on codes

In this section, we prove some general upper and lower bounds for  $X$ -admissible graphs in terms of the order of the graph, where  $X \in \text{CODES}$ . Several such general bounds have been established in the literature. We not only address them here but also extend them to some of the newer codes — like the OD-code, FD-code and the FTD-code — that have been introduced rather recently.

### 3.4.1 General upper bounds

If  $G$  is an  $X$ -admissible graph of order  $n$ , then an obvious upper bound for  $\gamma^X(G)$  is  $n$ . However, we show next that in some cases, we can do slightly better than this trivial upper bound.

#### 3.4.1.1 Codes based on domination

**Theorem 3.3** (Bondy [29]). *Let  $C$  be a set with  $|C| = n$  and let  $C_1, C_2, \dots, C_n$  be subsets of  $C$  for each  $i \in [n]$  such that  $C_i \neq C_j$  for each  $i, j \in [n]$  and  $i \neq j$ . Then, there exists an element  $x$  of  $S$  such that  $C_i \setminus \{x\} \neq C_j \setminus \{x\}$  for each  $i, j \in [n]$  and  $i \neq j$ .*

Recall that, in Lemma 3.3, we had stated a general version of Bondy's Theorem.

**Lemma 3.8.** *Let  $X \in \text{CODES}$  and let  $G_1$  and  $G_2$  be  $X$ -admissible graphs on disjoint vertex sets each of whose cardinalities is at least two. Let  $G = G_1 \oplus G_2$ . Moreover, let  $S_1$  be an  $X$ -code of  $G_1$ . Then, the set  $S = S_1 \cup V(G_2)$  is an  $X$ -code of  $G$ .*

*Proof.* Clearly, the set  $S$  has the same domination type as the set  $S_1$ . Let  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$ . Let  $u, v$  be two distinct vertices of  $G$ . Then, we show that the set  $S$  has non-empty intersection with the separator  $\Delta_A(G; u, v)$ . If  $u, v \in V(G_1)$ , then  $\Delta_A(G_1; u, v) = \Delta_A(G; u, v)$  and  $\Delta_A(G_1; u, v) \cap S_1 \neq \emptyset$  and hence,  $\Delta_A(G; u, v) \cap S \neq \emptyset$ . Now if  $u, v \in V(G_2)$ , then  $\Delta_A(G; u, v) = \Delta_A(G_2; u, v) \neq \emptyset$  since  $G_2$  is  $X$ -admissible. This implies that  $\Delta_A(G_2; u, v) \cap V(G_2) \neq \emptyset$  and hence,  $\Delta_A(G; u, v) \cap S \neq \emptyset$ . Therefore, let us assume that  $u \in V(G_1)$  and  $v \in V(G_2)$ . Since  $G_2$  is on at least two vertices, this implies that there exists a vertex  $v' \in N_{G_2}(v) \subseteq S$  with  $v' \neq v$ . Now, since  $\Delta_A(G; u, v) \in \{\Delta_O(G; u, v), \Delta_C(G; u, v)\}$ , we have either  $\Delta_A(G; u, v) = N_{G_1}[u] \cup N_{G_2}[v]$  or  $\Delta_A(G; u, v) = N_{G_1}(u) \cup N_{G_2}(v)$  (since  $G_1$  and  $G_2$  are disjoint). In either case, we have  $v' \in \Delta_A(G; u, v) \cap S$ . This proves the result.  $\square$

**Theorem 3.4.** *Let  $X \in \{LD, ID, OD\}$ . Then, for an  $X$ -admissible and isolate-free graph  $G$  on  $n \geq 2$  vertices, we have  $\gamma^X(G) \leq n - 1$ .*

*Proof.* For  $X = LD$ , the result is shown to be true in [186] and for  $X = ID$ , the result is shown to be true in [110]. Therefore we only need to prove the result for  $X = OD$  (also appears in [52] as part of the author's work). However, our proof also works equally well for  $X = ID$  and hence, let us assume that  $X \in \{ID, OD\}$ . Since  $G$  is without isolated vertices, it must have at least one edge. So, let  $G_1$  be a component of  $G$  with at least one edge and of order  $n_1$ . Then,  $G_1$  is also  $X$ -admissible. Let  $G_2$  be the disjoint union of all components of  $G$  other than  $G_1$ . In other words, we have  $G = G_1 \oplus G_2$ . Let  $S_1$  be a minimum  $X$ -code of  $G_1$ , that is,  $|S_1| = \gamma^X(G_1)$ . Now, if  $\gamma^X(G_1) \leq n_1 - 1$ , then, by applying Lemma 3.8, the set  $S = S_1 \cup V(G_2)$  is an  $X$ -code of  $G$ . This implies that  $\gamma^X(G_1) \leq |S| \leq n - 1$ .

Therefore, let us assume that  $\gamma^X(G_1) = |S_1| = n_1$ , that is,  $S_1 = V(G_1)$ . Now, let  $V(G_1) = \{v_1, v_2, \dots, v_{n_1}\}$ . Let us first assume that  $X = OD$ . Then, let  $C_i = N(v_i)$  for each  $i \in [n_1]$ . Since  $S_1$  is an  $OD$ -code of  $G_1$ , we have  $C_i \neq C_j$  for all  $i, j \in [n_1]$  and  $i \neq j$ . However, by Theorem 3.3 (Bondy's theorem), there exists a vertex  $x_1$  of  $G_1$  such that  $C_i \setminus \{x_1\} \neq C_j \setminus \{x_1\}$  for all  $i, j \in [n_1]$  and  $i \neq j$ . In other words, the set  $S_1 \setminus \{x_1\}$  is an open-separating set of  $G_1$ . Moreover,  $S_1 \setminus \{x_1\}$  is also a dominating set of  $G_1$ , since  $S_1 \setminus \{x_1\} = V(G_1) \setminus \{x_1\}$  and  $G_1$  is connected. Thus,  $S_1 \setminus \{x_1\}$  is an  $OD$ -code of  $G_1$  which is a contradiction to the minimality of  $S_1$  in being an  $OD$ -code of  $G_1$ . Therefore, we have  $\gamma^{OD}(G_1) \leq n_1 - 1$  and we are again done by our previous arguments.

On the other hand, if  $X = ID$ , then instead of  $N_G(v_i)$  as in the previous paragraph, we let  $C_i = N_G[v_i]$  for all  $i \in [n_1]$ . Then, by the exact same argument as in the case of  $X = OD$ , the result follows.  $\square$

#### 3.4.1.2 Codes based on total domination

Here we consider all graphs to be isolate-free or else a code based on total domination would not exist. On any isolate-free graph  $G$  on  $n$  vertices, any set of  $n - 1$  vertices is a total-dominating set of

$G$ . Moreover, since any set of  $n - 1$  vertices is also an L-separating set of  $G$ , it implies that any set of  $n - 1$  vertices is an LTD-code of  $G$ . This trivially implies that  $\gamma^{\text{LTD}}(G) \leq n - 1$ . It was also shown by Foucaud and Lehtilä in [101] that for any ITD-admissible graph  $G$ , we have  $\gamma^{\text{ITD}}(G) \leq n - 1$  except if  $G \cong P_3$  (in which case,  $\gamma^{\text{ITD}}(G) = n = 3$ ). The authors of [101] also characterize all ITD-admissible graphs on  $n$  vertices whose ITD-numbers equal  $n - 1$ .

On the other hand, it was shown by Foucaud et al. in [89] that there exist arbitrarily large OTD-admissible graphs  $G$  on  $n$  vertices for which  $\gamma^{\text{OTD}}(G) = n$ . In the same paper, the authors provide a full characterization of all OTD-admissible graphs which attain this  $n$ -upper bound and show that these extremal graphs are only those in the family of half-graphs. Moreover, by Theorem 3.2, we see that for any FTD-admissible graph  $G$  on  $n$  vertices, we have  $\gamma^{\text{FTD}}(G) \geq \gamma^{\text{OTD}}(G)$ . This implies that if  $G$  is an arbitrarily large half-graph, then also we have  $\gamma^{\text{FTD}}(G) = n$ .

## 3.4.2 General lower bounds

Given any  $X \in \text{CODES}$  and a positive integer, in this section, we find a general lower bound on  $\gamma^X(G)$  for all  $X$ -admissible graphs  $G$  in terms of the order, say  $n$ , of  $G$ . We find that such a general lower bound is logarithmic in  $n$ . For  $X \in \{\text{LD}, \text{LTD}, \text{ID}, \text{OTD}\}$ , these lower bounds had already been found in the literature (see, for example, [128, 164, 186, 190] for such results on LTD-, ID-, LD- and OTD-numbers, respectively). In this section, we generalize these individual proofs in the literature to a common proof technique to establish the logarithmic lower bounds in  $n$  of the  $X$ -numbers of graphs. Moreover, from our generalized proof, we also establish similar logarithmic lower bounds for  $X$ -numbers of graphs when  $X \in \{\text{ITD}, \text{OD}, \text{FD}, \text{FTD}\}$ . The latter results are new in the literature of identification problems.

Apart from the general lower bounds of  $X$ -numbers, we characterize the following two things.

- (1) Given any positive integer  $k$ , we characterize in terms of a common construction technique all  $X$ -admissible graphs  $G$  with  $\gamma^X(G) = k$ .
- (2) Given any positive integer  $k$ , we characterize all  $X$ -admissible graphs of the largest possible order such that  $\gamma^X(G) = k$ .

It is in fact as a result of the above two characterizations that the general logarithmic lower bound comes out as a consequence. Similar characterizations had been formulated in the literature for some individual codes. For example, in [164], Moncel provides similar characterization of all ID-admissible graphs on  $n$  vertices with  $X$ -numbers  $\lceil \log n \rceil$ , which happens to be the general lower bound for ID-admissible graphs on  $n$  vertices.

The results of this section are also available in [51].

### 3.4.2.1 An intuitive outline for characterizing all $X$ -admissible graphs with $X$ -number $k$

Let  $X \in \text{CODES}$  and let  $G$  be an  $X$ -admissible graph with  $X$ -number  $k$ . We then take a minimum  $X$ -code, say  $Z$ , of  $G$  and let  $H = G[Z]$ . Then, the graph  $H$  must be  $X$ -admissible as well. We now start with the graph  $H$  and build a *new*  $X$ -admissible graph — which we call by the name  $G_D$  — by introducing new vertices and edges to the already existing ones of  $H$ . The construction of the new graph  $G_D$  is formally laid out in the course of this section. The subscript  $D$  of  $G_D$  simply entails this construction process and its details, not essential to the current intuitive idea, will become clearer as we go on. We finally prove that our initial graph  $G$  is isomorphic to  $G_D$  thus characterizing  $G$  by having the same construction process as that of  $G_D$ . The construction of the new graph  $G_D$  is carried out as follows.

- (1) Apart from the vertices of  $H$ , we introduce all possible new vertices  $v$  (and edges incident with  $v$  and vertices of  $H$ ) such that each  $v$  has a unique neighborhood in  $H$ . We call this graph the

$\mathbb{I}^*$ -extension of  $H$  and denote it by  $G_{\mathbb{I}} = G_{\mathbb{I}}(H)$  (the entity  $\mathbb{I}^*$  represents a bit-representation technique and will be defined later).

- (2) The new graph  $G_{\mathbb{I}}$  may not be X-admissible. We now have to find a set of “critical vertices” (among the newly introduced ones) removing which from  $G_{\mathbb{I}}$  makes the graph X-admissible. These critical vertices depend both on the code X and the graph  $H$  we started with. Therefore, their set is called the  $(X, H)$ -critical set of  $G_{\mathbb{I}}$  and is denoted by  $D_X = D_X(H)$ . The graph obtained by removing this critical set from  $G_{\mathbb{I}}$  is called the  $\max(X)$ -extension of  $H$  and is denoted by  $G_X$ .
- (3) Removing from  $G_{\mathbb{I}}$  any superset of  $D_X$  and a subset of the newly introduced vertices also renders the graph X-admissible (and hence, the name “critical set” for  $D_X$  which must be the bare minimum to be removed to make the graph X-admissible). It turns out that one can choose a set  $D$  (a superset of  $D_X$ ) removing which produces an X-admissible graph  $G_D = G_{\mathbb{I}} - D$  that is of the same order as our original graph  $G$ . The graph  $G_D$  is called an  $X$ -extension of  $H$  — as opposed to a  $\max(X)$  extension of  $H$  defined in the previous point when  $D = D_X$ .
- (4) Finally, introducing some edges among the newly introduced vertices still remaining after the removal of  $D$  makes the two graphs  $G$  and  $G_D$  isomorphic to each other. This implies that any arbitrary X-admissible graph  $G$  with  $\gamma^X(G) = k$  is isomorphic to some X-extension of an X-admissible graph  $H$  on  $k$  vertices. This accounts for the first characterization as was stated before.
- (5) It turns out that the graphs  $G_X = G_{\mathbb{I}} - D_X$  are the maximum ordered X-admissible graphs with X-number  $k$ . This provides the second characterization as was stated before.

We now formally define the above concepts introduced in the intuitive outline and proceed with the section. Let  $G$  be a graph on  $n$  vertices, let  $S$  be a vertex subset of  $G$  such that  $|S| = k$  and let  $B \in \text{NBD-TYPE}$ . Moreover, let  $S = \{s_1, s_2, \dots, s_k\}$ . Define  $\mathbb{I} = \mathbb{I}^k = \{0, 1\}^k$  and the function  $\text{B-bit}(S; -) : V(G) \rightarrow \mathbb{I}$ , called the  $(B, S)$ -bit-string function on  $G$ , by the following rule: for any  $v \in V(G)$ , let the image of  $v$  under the function be  $\text{B-bit}(S; v) = (b_1, b_2, \dots, b_k)$ , where  $b_i = 1$  if  $s_i \in N_B(G; v) \cap S$  and  $b_i = 0$  if  $s_i \notin N_B(G; v) \cap S$ , for all  $i \in [k]$ . For any  $v \in V(G)$ , we call the  $k$ -tuple  $\text{B-bit}(S; v)$  the  $(B, S)$ -bit-string of  $v$ , or simply, the *bit-string* of  $v$  when the parameters  $B$  and  $S$  are well-understood from the context. Moreover, define the function  $\sigma_S : S \rightarrow \{0, 1\}^k$  by  $\sigma_S(u) = (b_1, b_2, \dots, b_k)$ , where  $b_i = 1$  if  $u = s_i$  for some  $i \in [n]$  and  $b_j = 0$  for all  $j \in [n]$  and  $j \neq i$ . Let  $\theta = (0, 0, \dots, 0) \in \mathbb{I}$ .

**Observation 3.3.** Let  $G$  be a graph on  $n$  vertices and  $S$  be any vertex subset of  $G$  such that  $|S| = k$ . Let  $A \in \text{SEP-TYPE}$  and  $T_G : \text{SEP-TYPE} \times V(G) \times V(G) \rightarrow \text{NBD-TYPE}$  be the type transformation function defined by Equation (2.1). Moreover, let  $u, v \in V(G)$  be distinct and  $T = T_G(A; u, v)$ .

- (1)  $\text{C-bit}(S; u) = \text{O-bit}(S; u)$  if  $u \notin S$  and  $\text{C-bit}(S; u) = \text{O-bit}(S; u) + \sigma_S(u)$  if  $u \in S$ . In particular, for  $u \in S$ ,  $\text{C-bit}(S; u) \neq \theta$ .
- (2) Let  $\bar{S} = V(G) - S$ . Then,  $\Delta_A(G; u, v) \cap S = \emptyset$  if and only if

$$\begin{aligned} T\text{-bit}(S; u) &\neq T\text{-bit}(S; v), & \text{for all } u, v \in S; \\ T\text{-bit}(S; u) &\neq \text{O-bit}(S; v), & \text{for all } u \in S, v \in \bar{S}; \\ \text{O-bit}(S; u) &\neq \text{O-bit}(S; v), & \text{for all } u, v \in \bar{S}. \end{aligned}$$

*Proof.* (1) Let  $S = \{s_1, s_2, \dots, s_k\}$ . By the definitions of closed and open neighborhoods of vertices, we have  $N_C(G; u) = N_O(G; u) + u$ . Thus, if  $u \notin S$ , we have  $N_C(G; u) \cap S = N_O(G; u) \cap S$  and therefore, by the definition of the bit-representation function, we have  $\text{C-bit}(S; u) = \text{O-bit}(S; u)$ . So, let  $u = s_j \in S$  for some  $j \in [n]$ . Then, for all  $i \in [n]$  and  $i \neq j$ , the  $i$ -th components of  $\text{C-bit}(S; u)$  and  $\text{O-bit}(S; u)$  are the same. Moreover, since  $u \notin N_O(G; u)$ , the  $j$ -th component of  $\text{O-bit}(S; u)$  is always 0. However, the  $j$ -th component of  $\text{C-bit}(S; u)$  is 1. This implies the result.

- (2)  $\Delta_A(G; u, v) \cap S = \emptyset$  for some distinct vertices  $u, v \in V(G)$  if and only if  $\Delta_T(G; u, v) \cap S = \emptyset$  by Remark 2.21. Moreover, since  $T \in \text{NBD-TYPE}$ , the latter equality follows if and only if

$N_T(G; u) \cap S = N_T(G; v) \cap S$ . Furthermore, by the definition of the bit-representation function, the last neighborhood equality holds if and only if  $T\text{-bit}(S; u) = T\text{-bit}(S; v)$ . Therefore, the result follows by using the fact from (1) that  $C\text{-bit}(S; u) = O\text{-bit}(S; u)$  for any vertex  $u \in \bar{S}$ .  $\square$

Let  $H$  be any graph on  $k$  vertices and let  $S = V(H)$ . Moreover, let  $\mathbb{I}^* = \mathbb{I} - \theta$ . We now build a supergraph  $G_{\mathbb{I}} = G_{\mathbb{I}}(H)$  of  $H$  by introducing some new vertices and edges by the following rule: for every  $\mathbf{b} \in \mathbb{I}^*$ , introduce a vertex  $v_{\mathbf{b}}$  and edges such that  $O\text{-bit}(S; v_{\mathbf{b}}) = \mathbf{b}$ . Let the graph  $G_{\mathbb{I}}$  be called an  $\mathbb{I}^*$ -extension of  $H$ . Let  $\bar{S}_{\mathbb{I}} = V(G_{\mathbb{I}}) \setminus S$ . Since  $v_{\mathbf{b}} \in \bar{S}_{\mathbb{I}}$ , by Observation 3.3(1), it implies that  $C\text{-bit}(S; v_{\mathbf{b}}) = \mathbf{b}$  as well. Then, by the definition of the bit-representation function, this implies that, for all  $U \subseteq S$  such that  $U \neq \emptyset$ , there exists a *unique* vertex  $v_U \in \bar{S}_{\mathbb{I}}$  such that  $N_O(G_{\mathbb{I}}; v_U) \cap S = U$ . Moreover, we let the edge subset  $E[G_{\mathbb{I}}; \bar{S}_{\mathbb{I}}, \bar{S}_{\mathbb{I}}]$  of  $G_{\mathbb{I}}$  to be any arbitrary non-reflexive and symmetric relation  $R$  on  $\bar{S}_{\mathbb{I}}$ . Then, we have

$$V(G_{\mathbb{I}}) = S \cup \{v_U : U \subseteq S \text{ and } U \neq \emptyset\}; \text{ and}$$

$$E(G_{\mathbb{I}}) = E(H) \cup R \cup \left( \bigcup_{U \subseteq S, U \neq \emptyset} \{uv_U : u \in U, v_U \in \bar{S}_{\mathbb{I}}\} \right),$$

where  $R$  is any non-reflexive relation on  $\bar{S}_{\mathbb{I}}$ .

**Remark 3.7.** Let  $H$  be a graph on  $k$  vertices,  $\mathbb{I} = \{0, 1\}^k$  and  $\mathbb{I}^* = \mathbb{I} - \theta$ . Moreover, let  $G_{\mathbb{I}}$  be an  $\mathbb{I}^*$ -extension of  $H$  such that  $G_{\mathbb{I}}$  is on  $n_{\mathbb{I}}$  vertices. Then, we have  $n_{\mathbb{I}} = 2^k - 1 + k$ .

Remark 3.7 shows that the order  $n_{\mathbb{I}}$  of  $G_{\mathbb{I}}$  is a function of  $k$  alone and does not depend on the relation  $R$  on  $\bar{S}_{\mathbb{I}}$ . In other words, we may write  $n_{\mathbb{I}} = n_{\mathbb{I}}(k)$  to stress the dependence on  $k$  whenever necessary. On the other hand, the graph  $G_{\mathbb{I}}$  does depend also on the relation  $R$  on  $\bar{S}_{\mathbb{I}}$  as far as its edge set is concerned. To also emphasize that fact, we denote the graph  $G_{\mathbb{I}}$  by  $G_{\mathbb{I}}(H, R)$  wherever necessary. Now, for  $A \in \text{SEP-TYPE}$ ,  $B \in \text{NBD-TYPE}$  and  $X \in \text{CODES}$  such that  $X \equiv (A, B)$ , assume  $H$  to be an  $X$ -admissible graph on  $k$  vertices. Moreover, define

$$D_X = D_X(H) = \{v \in \bar{S}_{\mathbb{I}} : \Delta_A(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}.$$

We call the set  $D_X$  the  $(X, H)$ -critical subset of  $G_{\mathbb{I}}$ . Now, let  $D$  be a vertex subset of  $G_{\mathbb{I}}$  such that  $D_X \subseteq D \subseteq \bar{S}_{\mathbb{I}}$ . Then, let the graph  $G_D = G_D(H, R_D) = G_{\mathbb{I}} - D$  be called an  $X$ -extension of  $H$ , where  $R_D = R \setminus E[G_{\mathbb{I}}; D, \bar{S}_{\mathbb{I}}]$ . Note that  $R_D$  is also a non-reflexive and symmetric relation on  $\bar{S}_{\mathbb{I}} \setminus D$ . Moreover, when  $D = D_X$ , we denote  $G_D$  by  $G_X$  and  $R_D$  by  $R_X$ ; and call the graph  $G_X = G_X(H, R_X)$  a  $\max(X)$ -extension of  $H$ . Let the graph  $G_D$  be on  $n_D$  vertices and let  $G_X$  be on  $n_X$  vertices. Then, we have  $n_D = n_{\mathbb{I}}(k) - |D|$  and  $n_X = n_{\mathbb{I}}(k) - |D_X(H)|$ . Moreover, as  $D_X \subseteq D$ , we have  $n_D \leq n_X$ . In particular, this implies that the order  $n_X$  of  $G_X$  depends only on  $X$  and  $H$  (note that  $k$  is the order of the graph  $H$ ). To emphasize this dependence wherever necessary, we may write  $n_X$  as  $n_X(H)$ . Thus, given an  $X \in \text{CODES}$  and an  $X$ -admissible graph  $H$ , the order of every  $\max(X)$ -extension of  $H$ , irrespective of the relation  $R_X$ , is the same  $n_X = n_X(H)$ .

**Lemma 3.9.** Let  $X \in \text{CODES}$  and let  $H$  be an  $X$ -admissible graph on  $k$  vertices. Moreover, let  $\mathbb{I} = \{0, 1\}^k$ , let  $\mathbb{I}^* = \mathbb{I} - \theta$ , let  $G_{\mathbb{I}}$  be an  $\mathbb{I}^*$ -extension of  $H$ , let  $D_X$  be the  $(X, H)$ -critical subset of  $G_{\mathbb{I}}$ , let  $D$  be a vertex subset of  $G_{\mathbb{I}}$  such that  $D_X \subseteq D \subseteq V(G_{\mathbb{I}}) \setminus V(H)$  and let  $G_D = G_{\mathbb{I}} - D$  be an  $X$ -extension of  $H$ . Then,  $G_D$  is an  $X$ -admissible graph and  $V(H)$  is an  $X$ -code of  $G_D$ . In particular,  $G_X$  is an  $X$ -admissible graph and  $V(H)$  is an  $X$ -code of  $G_X$ .

*Proof.* Let  $A \in \text{SEP-TYPE}$ ,  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$ . Moreover, let  $S = V(H)$  and  $\bar{S}_{\mathbb{I}} = V(G_{\mathbb{I}}) \setminus V(H)$ . Then, we have  $D_X = \{v \in \bar{S}_{\mathbb{I}} : \Delta_A(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$ .

The result follows by showing that  $S$  is an  $A$ -separating set of  $G_D$ . Since  $H$  is  $X$ -admissible by assumption, by Remark 2.22,  $S$  is an  $A$ -separating set of  $H$ . Therefore, we only need to show that  $\Delta_A(G_D; u, v) \cap S \neq \emptyset$  for all  $u \in S$  and  $v \in \bar{S}_D = \bar{S}_{\mathbb{I}} \setminus D$ ; and  $\Delta_A(G_D; v, v') \cap S \neq \emptyset$  for all distinct  $v, v' \in \bar{S}_D$ . Let us first assume that  $u \in S$  and  $v \in \bar{S}_D$ . Now, by definition, we create the graph  $G_D$  by deleting from  $G_{\mathbb{I}}$  at least the set  $D_X$  which contains exactly the vertices  $v \in \bar{S}_{\mathbb{I}}$  such that

$\Delta_A(G_{\mathbb{I}}; u, v) \cap S = \emptyset$  for some  $u \in S$ . Thus, by the definition of  $G_D$ , for all pairs of vertices  $u, v$  of  $G_D$  such that  $u \in S$  and  $v \in \bar{S}_D$ , we have  $\Delta_A(G_D; u, v) \cap S \neq \emptyset$ . Now, let  $v, v' \in \bar{S}_D$  be distinct. Since, in the construction of the graph  $G_{\mathbb{I}}$ , for each  $\mathbf{b} \in \mathbb{I}^*$ , a unique vertex  $v_{\mathbf{b}}$  is introduced in  $\bar{S}_{\mathbb{I}}$ , it implies that  $\text{O-bit}(S; v) \neq \text{O-bit}(S; v')$ . Therefore, by Observation 3.3(2), we have  $\Delta_A(G_{\mathbb{I}}; v, v') \cap S \neq \emptyset$ . This proves that the graph  $G_D$  is A-separable and that  $S$  is an A-separating set of  $G_D$ .

To show that  $G_X$  is X-admissible, we only need to show that  $G_D$  is isolate-free whenever  $B = O$ . So, on the contrary, let  $B = O$  and let  $G_D$  have an isolated vertex, say  $u$ . Since for every vertex  $v_{\mathbf{b}}$  in  $\bar{S}_{\mathbb{I}}$ , we have  $\mathbf{b} \neq \theta$ , it implies that  $v_{\mathbf{b}}$  is adjacent to some vertex in  $S$ . This further implies that the isolated vertex  $u$  must be in  $S$ . However, this also implies that  $u$  must be an isolated vertex of  $H$  which contradicts the fact that  $H$  is isolate-free on account of being X-admissible with  $X \equiv (A, O)$ . This proves that, indeed, the graph  $G_D$  is X-admissible.

We now show that  $S$  is an X-code of  $G_D$ . We have already shown that  $S$  is an A-separating set of  $G_D$ . Moreover, by the construction of  $G_D$ , the set  $S$  dominates every vertex of  $\bar{S}_D$ . In addition,  $S$  is also a dominating set of  $H$ . Hence,  $S$  is a dominating set of  $G_D$  and thus, is also an X-code of  $G_X$  if  $B = C$ . If  $B = O$ , being X-admissible, the graph  $H$  is isolate-free. Therefore,  $S$  is a total-dominating set of  $H$ . In other words,  $S$  is a total-dominating set of  $G_D$  and hence, is again an X-code of  $G_D$ .

Finally, taking  $D = D_X$ , this also proves that  $G_X$  is X-admissible and  $S$  is an X-code of  $G_X$ .  $\square$

**Lemma 3.10.** *Let  $A \in \text{SEP-TYPE}$ ,  $B \in \text{NBD-TYPE}$  and  $X \in \text{CODES}$  such that  $X \equiv (A, B)$ . Moreover, let  $H$  be an X-admissible graph on  $k$  vertices,  $\mathbb{I} = \{0, 1\}^k$ , let  $\mathbb{I}^* = \mathbb{I} - \theta$  and let  $G_{\mathbb{I}}$  be an  $\mathbb{I}^*$ -extension of  $H$ . Also, assume that  $S = V(H)$ ,  $\bar{S}_{\mathbb{I}} = V(G_{\mathbb{I}}) \setminus S$  and let  $D_X$  be the  $(X, H)$ -critical subset of  $G_{\mathbb{I}}$ . Then, the following are true.*

- (1) For  $A = L$ ,  $|D_X| = 0$ .
- (2) For  $A = C$ ,  $D_X = \{v \in \bar{S}_{\mathbb{I}} : C\text{-bit}(S; u) = O\text{-bit}(S; v), u \in S\}$ . In particular,  $|D_X| = k$ .
- (3) For  $A = O$ ,  $D_X = \{v \in \bar{S}_{\mathbb{I}} : O\text{-bit}(S; u) = O\text{-bit}(S; v), u \in S\}$ . In particular,  $|D_X| = k$  if  $H$  is isolate-free and  $|D_X| = k - 1$  if  $H$  has an isolated vertex.
- (4) For  $A = F$ ,  $D_X = \{v \in \bar{S}_{\mathbb{I}} : C\text{-bit}(S; u) = O\text{-bit}(S; v), u \in S\} \cup \{v \in \bar{S}_{\mathbb{I}} : O\text{-bit}(S; u) = O\text{-bit}(S; v), u \in S\}$ . In particular,  $|D_X| = 2k$  if  $H$  is isolate-free and  $|D_X| = 2k - 1$  if  $H$  has an isolated vertex.

*Proof.* By definition, we have  $D_X = \{v \in \bar{S}_{\mathbb{I}} : \Delta_A(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$ . Let us first look at  $A = L$ . Since  $\{u, v\} \subset \Delta_L(G_{\mathbb{I}}; u, v)$  for all  $u, v \in V(G_{\mathbb{I}})$ , therefore,  $u \in S$  implies that  $u \in \Delta_L(G_{\mathbb{I}}; u, v) \cap S$ . In other words,  $\Delta_L(G_{\mathbb{I}}; u, v) \cap S \neq \emptyset$  for all  $u \in S$  and  $v \in \bar{S}_{\mathbb{I}}$ . Thus,  $D_X = \emptyset$  and therefore,  $|D_X| = 0$ . This proves (1).

Let us now assume that  $A \in \text{SEP-TYPE} \setminus \{L\} = \{C, O, F\}$ . By Observation 3.3(2), we notice that

$$D_X = \{v \in \bar{S}_{\mathbb{I}} : T\text{-bit}(S; u) = O\text{-bit}(S; v), u \in S\}, \quad (3.4)$$

where  $T_{G_{\mathbb{I}}}$  is the type transformation function on  $G_{\mathbb{I}}$  and  $T = T_{G_{\mathbb{I}}}(A; u, v) \in \text{NBD-TYPE}$ . Let the set on the right hand side of Equation (3.4) be denoted by  $Y_T$ , where  $T \in \{C, O\}$ . We now count the order of the set  $Y_T$  by establishing a one-to-one correspondence between the vertices  $u \in S$  and  $v \in \bar{S}_{\mathbb{I}}$  such that  $T\text{-bit}(S; u) = O\text{-bit}(S; v)$ . To do so, we first notice that, since  $H$  is X-admissible, by Remark 2.22,  $S$  is an A-separating set of  $H$ . In other words, for any two distinct  $u, u' \in S$ , we have  $\Delta_A(S; u, u') \cap S \neq \emptyset$  which, by Observation 3.3(2), implies that  $T\text{-bit}(S; u) \neq T\text{-bit}(S; u')$ . This establishes a one-to-one correspondence between a vertex  $u \in S$  and its bit-string  $T\text{-bit}(S; u)$ .

First, let  $A = C$ . In this case,  $T = C$  as well. Then, by Observation 3.3(1),  $T\text{-bit}(S; u) \neq \theta$  for all  $u \in S$ . Therefore, by the construction of the graph  $G_{\mathbb{I}}$ , for each  $u \in S$ , there exists a unique vertex  $v \in \bar{S}_X$  such that  $T\text{-bit}(S; u) = O\text{-bit}(S; v)$ . This implies that  $|Y_T| = |S| = k$  and thus, proves (2).

Next, let  $A = O$ . Then,  $T = O$  as well. Again, by the construction of the graph  $G_{\mathbb{I}}$ , for each  $T\text{-bit}(S; u) \in \mathbb{I}^*$  (with  $u \in S$ ), there exists a unique vertex  $v \in \bar{S}_X$  such that  $T\text{-bit}(S; u) = O\text{-bit}(S; v)$ .

This implies that  $|Y_T| = |S| = k$  if  $\text{T-bit}(S; u) \neq \theta$  for all  $u \in S$ , or equivalently, if  $H$  is isolate-free; and  $|Y_T| = |S| - 1 = k - 1$  if  $\text{T-bit}(S; u_0) = \theta$  for some  $u_0 \in S$ , or equivalently, if  $H$  has an isolated vertex (which, by Remark 2.7, can be at most one, namely  $u_0$ ). This proves (3).

Finally, let  $A = F$ . In this case,  $T = C$  if  $uv \in E(G_{\mathbb{I}})$  and  $T = O$  if  $uv \notin E(G_{\mathbb{I}})$ , where  $u \in S$  and  $v \in \bar{S}$ . Therefore, we rewrite Equation (3.4) as follows.

$$D_X = \{v \in \bar{S} : \text{C-bit}(S; u) = \text{O-bit}(S; v), u \in S, uv \in E(G_{\mathbb{I}})\} \\ \bigcup \{v \in \bar{S} : \text{O-bit}(S; u) = \text{O-bit}(S; v), u \in S, uv \notin E(G_{\mathbb{I}})\}.$$

However, we notice that, for any  $u, v \in V(G_{\mathbb{I}})$  the condition  $\text{C-bit}(S; u) = \text{O-bit}(S; v)$  automatically implies that  $uv \in E(G_{\mathbb{I}})$  and, similarly, the condition  $\text{O-bit}(S; u) = \text{O-bit}(S; v)$  implies that  $uv \notin E(G_{\mathbb{I}})$ . Hence, the above expression for  $D_X$  can be simplified to the following.

$$D_X = \{v \in \bar{S} : \text{C-bit}(S; u) = \text{O-bit}(S; v), u \in S\} \\ \bigcup \{v \in \bar{S} : \text{O-bit}(S; u) = \text{O-bit}(S; v), u \in S\} = Y_C + Y_O. \quad (3.5)$$

We first show that the two sets on the right hand side of Equation (3.5) are disjoint. If on the contrary, there exists a vertex  $v \in \bar{S}_{\mathbb{I}}$  such that  $v$  belongs to both sets on the right hand side of Equation (3.5), then it implies  $\text{C-bit}(S; u) = \text{O-bit}(S; v) = \text{O-bit}(S; u')$  for some  $u, u' \in S$ . In other words, we have  $N_C(G_{\mathbb{I}}; u) = N_O(G_{\mathbb{I}}; u')$ . Since  $u \in N_C(G_{\mathbb{I}}; u)$  it implies  $uu' \in E(G_{\mathbb{I}})$ . However, since  $u' \notin N_O(G_{\mathbb{I}}; u')$ , it implies that  $uu' \notin E(G_{\mathbb{I}})$ . This leads to a contradiction. Hence, the two sets on the right hand side of Equation (3.5) are disjoint.

Now, using the cases when  $A = C$  and  $A = O$  from above, we have  $|D_F| = |Y_C| + |Y_O| = 2|S| = 2k$  if  $H$  is isolate-free and  $|D_F| = |Y_C| + |Y_O| = |S| + |S| - 1 = 2k - 1$  if  $H$  has an isolated vertex (which, again by Remark 2.7, can be at most one). This proves (4).  $\square$

**Lemma 3.11.** *Let  $X \in \text{CODES}$ , let  $H$  be an  $X$ -admissible graph on  $k$  vertices. Moreover, let  $\mathbb{I} = \{0, 1\}^k$ , let  $\mathbb{I}^* = \mathbb{I} - \theta$ , let  $G_{\mathbb{I}}$  be an  $\mathbb{I}^*$ -extension of  $H$  and let  $D_X$  be the  $(X, H)$ -critical subset of  $G_{\mathbb{I}}$ . In addition, let  $D$  be a vertex subset of  $G_{\mathbb{I}}$  such that  $D_X \subseteq D \subseteq V(G_{\mathbb{I}}) \setminus V(H)$ , let  $G_D = G_{\mathbb{I}} - D$  be an  $X$ -extension of  $H$  on  $n_D$  vertices and let  $G_X = G_{\mathbb{I}} - D_X$  be a  $\max(X)$ -extension of  $H$  on  $n_X$  vertices. Then, we have the following.*

$$n_D \leq n_X = \begin{cases} 2^k - 1 + k, & \text{if } X \in \{LD, LTD\} \\ 2^k, & \text{if } X = OD \text{ and } H \text{ has an isolated vertex} \\ 2^k - 1, & \text{if } X = OD \text{ and } H \text{ is isolate free} \\ 2^k - 1, & \text{if } X \in \{ID, ITD, OTD\} \\ 2^k - k, & \text{if } X = FD \text{ and } H \text{ has an isolated vertex} \\ 2^k - 1 - k, & \text{if } X = FD \text{ and } H \text{ is isolate-free} \\ 2^k - 1 - k, & \text{if } X = FTD. \end{cases} \quad (3.6)$$

In particular, we have

$$k \geq \begin{cases} \lfloor \log n_D \rfloor, & \text{if } X \in \{LD, LTD\} \\ \lceil \log n_D \rceil, & \text{if } X = OD \\ \lceil \log(n_D + 1) \rceil, & \text{if } X \in \{ID, ITD, OTD\} \\ 1 + \lfloor \log n_D \rfloor, & \text{if } X = FD \\ 1 + \lfloor \log(n_D + 1) \rfloor, & \text{if } X = FTD. \end{cases} \quad (3.7)$$

Moreover, for each positive integer  $k$ , there exists an  $X$ -admissible graph  $H$  on  $k$  vertices and an  $X$ -extension of  $H$ , say  $G_D$ , such that the equality in Equation (3.7) is attained.



*Proof.* Let  $X \equiv (A, B)$ , where  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$ . Also, let  $S = V(H)$  and  $\bar{S}_{\mathbb{I}} = V(G_{\mathbb{I}}) \setminus S$ . Then we have  $D_X = \{v \in \bar{S}_{\mathbb{I}} : \Delta_A(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$ . By Lemma 3.9, the graph  $G_D$  is  $X$ -admissible and  $S$  is an  $X$ -code of  $G_D$ . Moreover, by Remark 3.7, we have  $n_{\mathbb{I}} = 2^k - 1 + k$ . Therefore, we have  $n_D \leq n_X = n_{\mathbb{I}} - |D_X| = 2^k - 1 + k - |D_X|$ . Thus, the equality holds when  $D = D_X$ .

Let  $A = L$ . By Lemma 3.10(1), we have  $|D_X| = 0$ . This implies  $n_D \leq n_X = 2^k - 1 + k$ . This proves the result in Equation (3.6) for  $X \in \{\text{LD}, \text{LTD}\}$ . Now, we have  $k - 1 < 2^k$  for all  $k \in \mathbb{N}^*$ . Therefore,  $n_D \leq n_X < 2^{k+1}$ . This implies that  $\log n_D \leq \log n_X < k + 1$ . Hence, we have  $k \geq \lfloor \log n_X \rfloor \geq \lfloor \log n_D \rfloor$ . This proves Equation (3.7) for  $X \in \{\text{LD}, \text{LTD}\}$ . Moreover, we have  $2^k \leq 2^k - 1 + k = n_X$  for all  $k \in \mathbb{N}^*$  which implies that  $k \leq \log n_X$ , that is,  $k \leq \lfloor \log n_X \rfloor$ . Hence, we have  $k = \lfloor \log n_X \rfloor$  and the equality in Equation (3.7) for  $X \in \{\text{LD}, \text{LTD}\}$  is attained by taking  $D = D_X$ .

Let  $X = \text{OD} \equiv (A, B) = (O, C)$ . Then, by Lemma 3.10(3), we have  $|D_X| = |S| = k$  when  $H$  is isolate-free; and  $|D_X| = |S| - 1 = k - 1$  when  $H$  has an isolated vertex. Therefore, we have  $n_D \leq n_X = 2^k - 1$  when  $H$  is isolate-free; and  $n_D \leq n_X = 2^k$  when  $H$  has an isolated vertex. This proves the result in Equation (3.6) for  $X = \text{OD}$ . Therefore, we have  $n_D \leq n_X \leq 2^k$ . This implies that  $k \geq \log n_D$ , that is,  $k \geq \lfloor \log n_D \rfloor$ . This proves Equation (3.7) for  $X = \text{OD}$ . Now, let us take an  $X$ -admissible graph  $H$  on  $k$  vertices and with an isolated vertex. Then we have  $n_X = 2^k$  which implies that  $k = \log n_X = \lfloor \log n_X \rfloor$ . This proves that, for  $X = \text{OD}$ , there exists an  $X$ -admissible graph  $H$  for whose one of the  $X$ -extensions, namely the  $\max(X)$ -extension, the equality in Equation (3.7) holds.

Let  $A = C$ . By Lemma 3.10(2), we have  $|D_X| = |S| = k$ . This implies  $n_D \leq n_X = 2^k - 1$  and so, for  $X \in \{\text{ID}, \text{ITD}\}$ , we have  $n_D \leq n_X = 2^k - 1$ . Now, let  $X = \text{OTD} \equiv (A, B) = (O, O)$ . Then,  $H$  is isolate-free since  $H$  is  $\text{OTD}$ -admissible. Therefore, by Lemma 3.10(3), we have  $|D_X| = |S| = k$  and hence,  $n_D \leq n_X = 2^k - 1$ . In other words, for  $X \in \{\text{ID}, \text{ITD}, \text{OTD}\}$ , we have  $n_D \leq n_X = 2^k - 1$ . This proves the result in Equation (3.6) for  $X \in \{\text{ID}, \text{ITD}, \text{OTD}\}$ . This further implies that  $k = \log(n_X + 1) \geq \log(n_D + 1)$ , that is,  $k = \lceil \log(n_X + 1) \rceil \geq \lceil \log(n_D + 1) \rceil$ . This proves Equation (3.7) for  $X \in \{\text{ID}, \text{ITD}, \text{OTD}\}$  and its equality when  $D = D_X$ .

Let  $X = \text{FD} \equiv (A, B) = (F, C)$ . Then, by Lemma 3.10(4), we have  $|D_X| = 2|S| = 2k$  when  $H$  is isolate-free; and  $|D_X| = 2|S| - 1 = 2k - 1$  when  $H$  has an isolated vertex. Therefore, we have  $n_D \leq n_X = 2^k - 1 - k$  when  $H$  is isolate-free; and  $n_D \leq n_X = 2^k - k$  when  $H$  has an isolated vertex. This proves the result in Equation (3.6) for  $X = \text{FD}$ . This implies that  $n_D \leq n_X \leq 2^k - k$ . Now, since  $2^k - k < 2^k$  for all  $k \in \mathbb{N}^*$ , we have  $k > \log n_X \geq \log n_D$ , that is,  $k \geq 1 + \lfloor \log n_X \rfloor \geq 1 + \lfloor \log n_D \rfloor$ . This proves Equation (3.7) for  $X = \text{FD}$ . Moreover, we have  $2^{k-1} \leq 2^k - k$  for all  $k \in \mathbb{N}^*$ . We now take an  $X$ -admissible graph  $H$  on  $k$  vertices that has an isolated vertex. Then, we have  $2^{k-1} \leq 2^k - k = n_X$  which implies  $k \leq 1 + \log n_X$ , that is,  $k \leq 1 + \lfloor \log n_X \rfloor$ . In other words, we have  $k = 1 + \lfloor \log n_X \rfloor$ . Thus, for  $X = \text{FD}$ , there exists an  $X$ -admissible  $H$  for whose one of the  $X$ -extensions, namely the  $\max(X)$ -extension, the equality in Equation (3.7) holds.

Let  $X = \text{FTD} \equiv (A, B) = (F, O)$ . This implies that  $H$  is isolate-free, since  $H$  is  $\text{FTD}$ -admissible. Therefore, by Lemma 3.10(4), we have  $|D_X| = 2|S| = 2k$  and so,  $n_D \leq n_X = 2^k - 1 - k$ . This proves the result in Equation (3.6) for  $X = \text{FTD}$ . Now, since  $2^k - 1 - k < 2^k - 1$  for all  $k \in \mathbb{N}^*$ , we have  $k > \log(n_X + 1) \geq \log(n_D + 1)$ , that is,  $k \geq 1 + \lfloor \log(n_X + 1) \rfloor \geq 1 + \lfloor \log(n_D + 1) \rfloor$ . This proves Equation (3.7) for  $X = \text{FTD}$ . Moreover, we have  $2^{k-1} - 1 \leq 2^k - 1 - k = n_X$  for all  $k \in \mathbb{N}^*$  which implies  $k \leq 1 + \log(n_X + 1)$ , that is,  $k \leq 1 + \lfloor \log(n_X + 1) \rfloor$ . In other words,  $k = 1 + \lfloor \log(n_X + 1) \rfloor$  and this proves the equality in Equation (3.7) for  $X = \text{FTD}$ .  $\square$

**Lemma 3.12.** *Let  $X \in \text{CODES}$  and let  $G$  be an  $X$ -admissible graph on  $n$  vertices. Moreover, let  $Z \subseteq V(G)$  be an  $X$ -code of  $G$  and let  $H$  be a graph isomorphic to  $G[Z]$ . Then the graph  $G$  is isomorphic to an  $X$ -extension of  $H$ . In particular, if  $G_X$  is a  $\max(X)$ -extension of  $H$  on  $n_X$  vertices, then we have  $n \leq n_X$ .*

*Proof.* Let  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$  such that  $X \equiv (A, B)$ . Also, let  $|Z| = k$ , let  $J = G[Z]$ , let  $\iota : J \rightarrow H$  be an isomorphism and let  $S = V(H)$ . Moreover, let  $\mathbb{I} = \{0, 1\}^k$ ,  $\mathbb{I}^* = \mathbb{I} - \theta$

and  $G_{\mathbb{I}} = G_{\mathbb{I}}(H, R)$  be an  $\mathbb{I}^*$ -extension of  $H$ , where  $R$  is some non-reflexive and symmetric relation on  $\bar{S}_{\mathbb{I}} = V(G_{\mathbb{I}}) \setminus S$ . Therefore, we have  $G_X = G_X(H, R_X) = G_{\mathbb{I}} - D_X$ , where  $D_X = \{v \in \bar{S}_{\mathbb{I}} : \Delta_A(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$  and  $R_X = R \setminus E[D_X, \bar{S}_{\mathbb{I}}]$ . We shall address later in the proof what the set  $R_X$  is. We now define a function  $f : V(G) \rightarrow V(G_X)$  according to the following analysis which does not depend on  $R_X$ .

If  $z \in Z$ , we let  $f(z) = \iota(z)$ . Now, let  $\bar{Z} = V(G) \setminus Z$  and let  $y \in \bar{Z}$ . We now look for a vertex  $v \in \bar{S}_X = \bar{S}_{\mathbb{I}} \setminus D_X$  so that we can define  $f(y) = v$ . Since  $Z$  is an  $X$ -code of  $G$  and  $y \notin Z$ , there exists a neighbor of  $y$  in  $Z$ . This implies that  $\text{O-bit}(Z; y) \neq \theta$ . Therefore, by the construction of the graph  $G_{\mathbb{I}}$ , there exists a vertex  $v \in \bar{S}_{\mathbb{I}}$  such that

$$\text{O-bit}(S; v) = \text{O-bit}(Z; y).$$

We shall finally show that  $v \in \bar{S}_X$ , that is,  $v \notin D_X$ . However, we first argue that if  $z \in Z$  and  $u \in S$  such that  $\iota(z) = u$ , then  $zy \in E(G)$  if and only if  $uv \in E(G_{\mathbb{I}})$ . To do so, let  $Z = \{z_1, z_2, \dots, z_k\}$  and  $S = \{u_1, u_2, \dots, u_k\}$  such that  $\iota(z_i) = u_i$  for all  $i \in [k]$ . Therefore, if  $z = z_j$  for some  $j \in [k]$ , we have  $u = u_j$ . Since  $\text{O-bit}(S; v) = \text{O-bit}(Z; y)$ , the  $j$ -th components of both the bit-strings are the same. This proves that  $zy \in E(G) \iff uv \in E(G_{\mathbb{I}})$ .

We now show that  $v \notin D_X$ . Assume on the contrary that  $v \in D_X$  and so, there exists a vertex  $u \in S$  such that  $\Delta_A(G_{\mathbb{I}}; u, v) \cap S = \emptyset$ . Since  $u \in S$  and  $v \in \bar{S}_{\mathbb{I}}$ , by Observation 3.3(2), we have

$$\text{T-bit}(S; u) = \text{O-bit}(S; v),$$

where  $T_{G_{\mathbb{I}}}$  is the type transformation function on  $G_{\mathbb{I}}$  and  $T = T_{G_{\mathbb{I}}}(A; u, v)$ . Now, let  $z = \iota^{-1}(u)$ . Since, as shown before,  $zy \in E(G) \iff uv \in E(G_{\mathbb{I}})$ , we therefore have  $T_G(A; z, y) = T_{G_{\mathbb{I}}}(A; u, v) = T$ , where  $T_G$  is the type transformation function on  $G$ . Moreover, since  $\iota$  is an isomorphism, we also have

$$\text{T-bit}(Z; z) = \text{T-bit}(S; u).$$

Thus, combining the above three bit string equations, we get  $\text{T-bit}(Z; z) = \text{O-bit}(Z; y)$ . However, since  $z \in Z$  and  $y \in \bar{Z}$ , by Observation 3.3(2), we have  $\Delta_A(G; z, y) \cap Z = \emptyset$ . This contradicts the fact that  $Z$  is an  $X$ -code of  $G$ . This implies that  $v \notin D_X$ . This proves that  $v \in \bar{S}_X \subseteq V(G_X)$ . Finally then, we define  $f(y) = v$ .

We now show that the function  $f$  is injective. Since  $\iota$  is already an isomorphism,  $f$  restricted to  $Z$  is an injective function, namely, the isomorphism  $\iota$ . Moreover, since for every  $z \in Z$  and  $y \in \bar{Z}$ , we have  $f(z) \in S$  and  $f(y) \in \bar{S}_X = V(G_X) \setminus S$ , it implies that  $f(y) \neq f(z)$ . Therefore, to show that  $f$  is injective, we only need to establish a one-to-one correspondence between each  $y \in \bar{Z}$  and a vertex  $v = f(y) \in \bar{S}_X$ , that is, for which we have  $\text{O-bit}(S; v) = \text{O-bit}(Z; y)$ . To do so, we notice that, since  $Z$  is an  $X$ -code and hence, an  $A$ -separating set of  $G$ , for each pair of distinct vertices  $y, y' \in \bar{Z}$ , by Observation 3.3(2), we have  $\text{O-bit}(Z; y) \neq \text{O-bit}(Z; y')$ . Now, if  $v, v' \in \bar{S}_X$  such that  $\text{O-bit}(S; v) = \text{O-bit}(Z; y)$  and  $\text{O-bit}(S; v') = \text{O-bit}(Z; y')$ , then, we have  $\text{O-bit}(S; v) \neq \text{O-bit}(S; v')$ . This proves the said one-to-one correspondence and hence, the function  $f$  is injective.

We now let the set  $R = \{f(v)f(v') : v, v' \in \bar{Z} \text{ and } vv' \in E(G)\}$  and define the set  $D = V(G_{\mathbb{I}}) \setminus f(V(G))$ . Then the graph  $G_D = G_{\mathbb{I}} - D$  is an  $X$ -extension of  $H$  that is the image of the injective homomorphism  $f$ . Hence,  $G \cong G_D$  and this proves the result.  $\square$

**Lemma 3.13.** *Let  $X \in \text{CODES}$ , let  $H$  be an  $X$ -admissible graph on  $k$  vertices and let  $G_X$  be a  $\max(X)$ -extension of  $H$ . Then,  $G_X$  is  $X$ -admissible and  $\gamma^X(G_X) = k$ . Moreover,  $V(H)$  is a minimum  $X$ -code of  $G_X$ .*

*Proof.* Let the graph  $G_X$  be on  $n_X$  vertices and  $S = V(H)$ . Then, by Lemma 3.9,  $G_X$  is  $X$ -admissible and the set  $S$  is an  $X$ -code of  $G_X$ . This implies that  $\gamma^X(G_X) \leq |S| = k$ . To prove  $\gamma^X(G_X) = k$ , let us assume on the contrary that  $k \geq 2$  and that  $\gamma^X(G_X) \leq k - 1$ . In other words, there exists an  $X$ -code  $S'$  of  $G_X$  such that  $1 \leq |S'| \leq k - 1$ . By possibly including more vertices in the set

$S'$ , let us assume without loss of generality that  $|S'| = k - 1$ . Then  $S'$  is still an  $X$ -code of  $G_X$ . Also, let  $H' = G[S']$ . Then, by Lemma 3.12, the graph  $G_X$  is isomorphic to an  $X$ -extension of  $H'$ , say  $G'$  on  $n'$  vertices. This implies that  $n_X = n'$ . Now, by Equation (3.6) in Lemma 3.11, we have  $n' \leq 2^{k-1} + \alpha_{X,H'}(k-1) + \beta_{X,H'}$ , where  $\alpha_{X,H'} \in \{-1, 0, 1\}$  and  $\beta_{X,H'} \in \{-1, 0\}$ . Moreover, since  $G_X$  is a  $\max(X)$ -extension of  $H$ , again by Equation (3.6) in Lemma 3.11, it implies that  $n_X = 2^k + \alpha_{X,H}k + \beta_{X,H}$ .

Notice that  $\alpha_{X,H'} = \alpha_{X,H}$  since the coefficients of the parameter  $k$  in the Equation (3.6) are independent of whether one of  $H$  and  $H'$  has an isolated vertex and the other one does not. On the other hand, we have  $\beta_{X,H'} = \beta_{X,H} + \epsilon$ , where  $\epsilon \in \{-1, 0, 1\}$  depends on whether one of  $H$  and  $H'$  has an isolated vertex and the other one does not. Moreover,  $\epsilon = 0$  for  $X \in \text{CODES} \setminus \{\text{OD}, \text{FD}\}$ . Therefore, we have

$$\begin{aligned} n' &\leq 2^{k-1} + \alpha_{X,H'}(k-1) + \beta_{X,H'} \\ &= 2^{k-1} + \alpha_{X,H}(k-1) + \beta_{X,H} + \epsilon \\ &= n_X + 2^{k-1} - 2^k + \epsilon - \alpha_{X,H} \\ &\leq n_X + 2^{k-1} - 2^k + \epsilon + 1. \end{aligned}$$

Therefore, we have  $n' < n_X$  for all  $k \geq 3$  and for  $k = 2$  with  $X \in \text{CODES} \setminus \{\text{OD}, \text{FD}\}$ . This is a contradiction to our conclusion  $n_X = n'$  earlier. Therefore, we have  $\gamma^X(G_X) \geq k$  and so,  $\gamma^X(G_X) = k$  for all  $k \geq 3$  and  $k = 2$  with  $X \in \text{CODES} \setminus \{\text{OD}, \text{FD}\}$ . Let us now assume that  $k = 2$  and that  $X \in \{\text{OD}, \text{FD}\}$ . Then, we have either  $H \cong P_2$  or  $H \cong 2K_1$ . In either case,  $H$  is not twin-free and hence is not FD-admissible. Therefore, for  $X = \text{FD}$ , we must have  $k \geq 3$  and hence,  $\gamma^X(G_X) = k$  by our earlier argument. Hence, let  $k = 2$  and  $X = \text{OD}$ . In this case, if  $H \cong 2K_1$ , it has two isolated vertices. Therefore, by Remark 2.7,  $H$  is not open-separable and hence, is not OD-admissible. Thus, we have  $H \cong P_2$  and so,  $|S| = 2$  and  $|S'| = 1$ . This implies that  $n_X = 3$  by Equation (3.6). Moreover,  $|S'| = 1$  implies that  $H'$  has an isolated vertex and so, by Equation (3.6), we have  $n' \leq 2 < 3 = n_X$ , again a contradiction. Hence, for  $X = \text{OD}$  as well, we must have  $k \geq 3$  and therefore,  $\gamma^X(G_X) = k$  by our earlier argument.  $\square$

**Theorem 3.5.** *Let  $X \in \text{CODES}$  and let  $G$  be an  $X$ -admissible graph on  $n$  vertices and with  $X$ -number  $k$ . Then,  $G$  is isomorphic to an  $X$ -extension of  $G[Z]$ , where  $Z$  is a minimum  $X$ -code of  $G$ . Moreover, we have*

$$n \leq \begin{cases} 2^k - 1 + k, & \text{if } X \in \{LD, LTD\} \\ 2^k, & \text{if } X = OD \\ 2^k - 1, & \text{if } X \in \{ID, ITD, OTD\} \\ 2^k - k, & \text{if } X = FD \\ 2^k - 1 - k, & \text{if } X = FTD. \end{cases} \quad \text{and} \quad (3.8)$$

$$\gamma^X(G) = k \geq \begin{cases} \lfloor \log n \rfloor, & \text{if } X \in \{LD, LTD\} \\ \lfloor \log n \rfloor, & \text{if } X = OD \\ \lfloor \log(n+1) \rfloor, & \text{if } X \in \{ID, ITD, OTD\} \\ 1 + \lfloor \log n \rfloor, & \text{if } X = FD \\ 1 + \lfloor \log(n+1) \rfloor, & \text{if } X = FTD. \end{cases} \quad (3.9)$$

In addition, for each integer  $k$ , there exist graphs with  $X$ -number  $k$  whose orders attain the upper bound in Equation (3.8).

*Proof.* Let  $Z \subseteq V(G)$  be a minimum  $X$ -code of  $G$ . Moreover, let  $H = G[Z]$ . Then, by Lemma 3.11, the graph  $G$  is isomorphic an  $X$ -extension of  $H$ .

Let  $\mathbb{I} = \{0, 1\}^k$ , let  $\mathbb{I}^* = \mathbb{I} - \theta$ , let  $G_{\mathbb{I}}$  be an  $\mathbb{I}^*$ -extension of  $H$  and let  $D_X$  be the  $(X, H)$ -critical subset of  $G_{\mathbb{I}}$ . In addition, let  $D$  be a vertex subset of  $G_{\mathbb{I}}$  such that  $D_X \subseteq D \subseteq V(G_{\mathbb{I}}) \setminus V(H)$  and

let  $G_D = G_{\mathbb{I}} - D$  be an  $X$ -extension of  $H$  on  $n_D$  vertices isomorphic to the graph  $G$ . Then, we have  $n_D = n$  and  $\gamma^X(G_D) = \gamma^X(G) = k$ . Notice that, by the Equation (3.6) in Lemma 3.11, the order  $n_D$  of  $G_D$  is at most the right hand side of Equation (3.8). Therefore, the result in Equation (3.8) holds by the fact that  $n = n_D$ . Similarly, using  $n = n_D$  in Equation (3.7) in Lemma 3.11 and the fact that  $\gamma^X(G) = \gamma^X(G_D) = k$ , the result in Equation (3.9) holds as well.

For each integer  $k$ , we choose an  $X$ -admissible graph  $H$  and take a  $\max(X)$ -extension of  $H$ , say  $G_X$  on  $n_X$  vertices (for  $X \in \{\text{OD}, \text{FD}\}$ , we choose  $H$  to have an isolated vertex). Then by Lemma 3.11, the order  $n_X$  equals the upper bound in Equation (3.8). Moreover, each such graph  $G_X$  has  $X$ -number  $k$  by Lemma 3.13.  $\square$

### 3.4.2.2 Common graph families whose members attain the general logarithmic lower bound

Given any  $X \in \text{CODES}$ , varying the graph  $H$  in a  $\max(X)$ -extension of  $H$  and the edge subset  $R_X$  (which could have been chosen to be any symmetric relation on  $\bar{S}_{\mathbb{I}} \setminus D_X$ ), we obtain several graphs whose  $X$ -numbers attain the lower bounds in Theorem 3.5. The results in this section point out examples of such extremal graphs from different graph families. For other known extremal graphs from other graph families, we refer the reader to consult Tables 2.1 to 2.10.

**Proposition 3.1.** *For each positive integer  $k$ , there exists a graph, say  $G$ , on  $n$  vertices from each of the families of bipartite graphs, cobipartite graphs, split graphs and chordal graphs such that  $\gamma^{\text{LD}}(G) = \lfloor \log n \rfloor = k$ .*

*Proof.* Let  $H$  be a graph on  $k$  vertices and  $S = V(H)$ . Let  $\mathbb{I} = \{0, 1\}^k$ ,  $\mathbb{I}^* = \mathbb{I} - \theta$  and  $G_{\mathbb{I}} = G_{\mathbb{I}}(H, R)$  be an  $\mathbb{I}^*$ -extension of  $H$ , where  $R$  is some non-reflexive relation on  $\bar{S} = V(G_{\mathbb{I}}) \setminus S$ . Moreover, let  $G_{\text{LD}} = G_{\text{LD}}(H, R_{\text{LD}}) = G_{\mathbb{I}} - D_{\text{LD}}$  be a  $\max(\text{LD})$ -extension of  $H$ , where  $D_{\text{LD}} = \{v \in \bar{S} : \Delta_{\mathbb{I}}(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$  and  $R_{\text{LD}} = R \setminus E[D_{\text{LD}}, \bar{S}]$ .

We next choose the graph  $H$  and the relation  $R$  so that  $G = G_{\text{LD}}$  is either a bipartite graph or a cobipartite graph or a split graph or a chordal graph. For  $G$  to belong to the family of either cobipartite or split graphs, we take a graph  $H \cong K_k$ . If  $G$  is to be a cobipartite graph, the result holds by taking  $R = \bar{S} \times \bar{S} \setminus \{(v, v) : v \in \bar{S}\}$ . For  $G$  to be a split graph, the result holds by taking  $R = \emptyset$ . Since spit graphs are chordal graphs, the result also holds for chordal graphs. Finally, for  $G$  to be a bipartite graph, we let  $H \cong \bar{K}_k$  and  $R = \emptyset$ .  $\square$

**Proposition 3.2.** *For each positive integer  $k \geq 2$ , there exists a graph, say  $G$ , on  $n$  vertices from each of the families of cobipartite graphs, split graphs and chordal graphs such that  $\gamma^{\text{LTD}}(G) = \lfloor \log n \rfloor = k$ .*

*Proof.* Let  $H$  be a graph on  $k$  vertices and  $S = V(H)$ . Let  $\mathbb{I} = \{0, 1\}^k$ ,  $\mathbb{I}^* = \mathbb{I} - \theta$  and  $G_{\mathbb{I}} = G_{\mathbb{I}}(H, R)$  be an  $\mathbb{I}^*$ -extension of  $H$ , where  $R$  is some non-reflexive relation on  $\bar{S} = V(G_{\mathbb{I}}) \setminus S$ . Moreover, let  $G_{\text{LTD}} = G_{\text{LTD}}(H, R_{\text{LTD}}) = G_{\mathbb{I}} - D_{\text{LTD}}$  be a  $\max(\text{LTD})$ -extension of  $H$ , where  $D_{\text{LTD}} = \{v \in \bar{S} : \Delta_{\mathbb{I}}(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$  and  $R_{\text{LTD}} = R \setminus E[D_{\text{LTD}}, \bar{S}]$ .

We next choose the graph  $H$  and the relation  $R$  so that  $G = G_{\text{LTD}}$  is either a cobipartite graph or a split graph or a chordal graph. For  $G$  to belong to the family of either cobipartite or split graphs, we take a graph  $H \cong K_k$ . For  $G$  to be a cobipartite graph, the result holds by taking  $R = \bar{S} \times \bar{S} \setminus \{(v, v) : v \in \bar{S}\}$ . If  $G$  is to be a split graph, the result holds by taking  $R = \emptyset$ . Since split graphs are chordal graphs, the result also holds for chordal graphs.  $\square$

**Proposition 3.3.** *For each positive integer  $k$ , there exists a split graph, say  $G$ , on  $n$  vertices such that  $\gamma^{\text{ID}}(G) = \lfloor \log(n+1) \rfloor = k$ .*

*Proof.* Let  $H$  be a graph on  $k$  vertices and  $S = V(H)$ . Let  $\mathbb{I} = \{0, 1\}^k$ ,  $\mathbb{I}^* = \mathbb{I} - \theta$  and  $G_{\mathbb{I}} = G_{\mathbb{I}}(H, R)$  be an  $\mathbb{I}^*$ -extension of  $H$ , where  $R$  is some non-reflexive relation on  $\bar{S} = V(G_{\mathbb{I}}) \setminus S$ . Moreover, let

$G_{\text{ID}} = G_{\text{ID}}(H, R_{\text{ID}}) = G_{\mathbb{I}} - D_{\text{ID}}$  be a  $\max(\text{ID})$ -extension of  $H$ , where  $D_{\text{ID}} = \{v \in \bar{S} : \Delta_{\text{C}}(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$  and  $R_{\text{ID}} = R \setminus E[D_{\text{ID}}, \bar{S}]$ .

We next choose the graph  $H$  and the relation  $R$  so that  $G = G_{\text{ID}}$  is a split graph. This is achieved by taking  $H \cong \overline{K_k}$  and  $R = \bar{S} \times \bar{S} \setminus \{(v, v) : v \in \bar{S}\}$ .  $\square$

**Proposition 3.4.** *For each positive integer  $k \geq 2$ , there exists a graph, say  $G$ , on  $n$  vertices from each of the families of cobipartite graphs, split graphs and chordal graphs such that  $\gamma^{\text{OD}}(G) = \lceil \log n \rceil = k$ .*

*Proof.* Let  $H$  be a graph on  $k$  vertices and  $S = V(H)$ . Let  $\mathbb{I} = \{0, 1\}^k$ ,  $\mathbb{I}^* = \mathbb{I} - \theta$  and  $G_{\mathbb{I}} = G_{\mathbb{I}}(H, R)$  be an  $\mathbb{I}^*$ -extension of  $H$ , where  $R$  is some non-reflexive relation on  $\bar{S} = V(G_{\mathbb{I}}) \setminus S$ . Moreover, let  $G_{\text{OD}} = G_{\text{OD}}(H, R_{\text{OD}}) = G_{\mathbb{I}} - D_{\text{OD}}$  be a  $\max(\text{OD})$ -extension of  $H$ , where  $D_{\text{OD}} = \{v \in \bar{S} : \Delta_{\text{O}}(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$  and  $R_{\text{OD}} = R \setminus E[D_{\text{OD}}, \bar{S}]$ .

We next choose the graph  $H$  and the relation  $R$  so that  $G = G_{\text{OD}}$  is either a cobipartite graph or a split graph or a chordal graph. We first take  $H \cong K_k$ . Since  $H$  is isolate-free, by Lemma 3.11, we have  $n = 2^k - 1$ . Now, since  $k \geq 2$ , we have  $2^{k-1} < 2^k - 1 < 2^k$ , that is,  $2^{k-1} < 2^k - 1 < 2^k$ . Hence, we have  $k - 1 < \log n < k$  which implies that  $k = \lceil \log n \rceil$ .

Then, the result for  $G$  cobipartite holds by taking  $R = \bar{S} \times \bar{S} \setminus \{(v, v) : v \in \bar{S}\}$ . If  $G$  is to be a split graph, the result holds by taking  $R = \emptyset$ . Since split graphs are chordal graphs, the result also holds for chordal graphs.  $\square$

**Proposition 3.5.** *For each positive integer  $k \geq 2$ , there exists a graph, say  $G$ , on  $n$  vertices from each of the families of cobipartite graphs, split graphs and chordal graphs such that  $\gamma^{\text{OTD}}(G) = \lceil \log(n + 1) \rceil = k$ .*

*Proof.* Let  $H$  be a graph on  $k$  vertices and  $S = V(H)$ . Let  $\mathbb{I} = \{0, 1\}^k$ ,  $\mathbb{I}^* = \mathbb{I} - \theta$  and  $G_{\mathbb{I}} = G_{\mathbb{I}}(H, R)$  be an  $\mathbb{I}^*$ -extension of  $H$ , where  $R$  is some non-reflexive relation on  $\bar{S} = V(G_{\mathbb{I}}) \setminus S$ . Moreover, let  $G_{\text{OTD}} = G_{\text{OTD}}(H, R_{\text{OTD}}) = G_{\mathbb{I}} - D_{\text{OTD}}$  be a  $\max(\text{OTD})$ -extension of  $H$ , where  $D_{\text{OTD}} = \{v \in \bar{S} : \Delta_{\text{O}}(G_{\mathbb{I}}; u, v) \cap S = \emptyset, u \in S\}$  and  $R_{\text{OTD}} = R \setminus E[D_{\text{OTD}}, \bar{S}]$ .

We next choose the graph  $H$  and the relation  $R$  so that  $G = G_{\text{OTD}}$  is either a cobipartite graph or a split graph or a chordal graph. We first take  $H \cong K_k$ . Then, the result for  $G$  cobipartite holds by taking  $R = \bar{S} \times \bar{S} \setminus \{(v, v) : v \in \bar{S}\}$ . If  $G$  is to be a split graph, the result holds by taking  $R = \emptyset$ . Since split graphs are chordal graphs, the result also holds for chordal graphs.  $\square$

## 3.5 Conclusion

In this chapter, we tried to look at all the eight codes from a holistic point of view. It involves reformulating the problems of finding codes in graphs to finding covers of hypergraphs. This also gives a way to make use of the incidence matrices of these hypergraphs to adopt a polyhedral approach to minimizing the order of codes on graphs. Using some of these reformulated tools, as depicted in Figure 3.2, we have been able to establish a comprehensive pairwise comparisons between the various X-codes.

In [116] and in [194] where the relations between LD-numbers, ID-numbers and OTD-numbers were studied, it was shown that Questions 3.1 could be answered for these codes with  $\alpha \leq 2$ . Theorem 3.2 is a generalization of their results and holds for other code comparisons not yet available in the literature. However, Corollary 3.7 shows that between the FTD-number and the LD-number such a relation may require a factor of  $\alpha = 4$ . We believe that in this case as well, the value of  $\alpha$  would be at most 2. To that end, we make the following conjecture.

**Conjecture 3.1.** *Let  $G$  be an FTD-admissible graph. Then, we have  $\gamma^{\text{FTD}}(G) \leq 2\gamma^{\text{LD}}(G) + 2$ .*

We have also looked at the most general upper and lower bounds of the code numbers. Combining with some known results from the literature, we have been able to put a general upper bound of  $n - 1$  on all domination-based codes of isolate-free graphs, except for FD-codes. However, we believe that this should be the case for FD-codes as well. As a result, we hereby make the following conjecture.

**Conjecture 3.2.** *Let  $G$  be an isolate-free FD-admissible graph on  $n \geq 4$  vertices. Then, we have  $\gamma^{\text{FD}}(G) \leq n - 1$  except when  $G \cong P_4$  in which case,  $\gamma^{\text{FD}}(G) = 4$ .*

Our general (logarithmic) lower bound result encapsulates some of the known such lower bounds and characterizations of graphs whose code numbers attain these lower bounds. In addition, by our general approach to the result, we have been able to establish equivalent lower bounds for the X-numbers with  $X \in \{\text{ITD}, \text{OD}, \text{FD}, \text{FTD}\}$ .



## Part I

# Structural aspects: Bounds on code numbers





# Chapter 4

## Location in graphs

In this chapter, we look at location in graphs, that is, we first study locating dominating codes in Section 4.1; and then in Section 4.2, we look at locating total-dominating codes in graphs. In particular, in both the sections, we look at location in twin-free and isolate-free graphs. This invokes the study of the  $n$ -half upper bound conjecture (that is, Conjecture 2.2 in Chapter 2) by Garijo et al. [112] on the LD-numbers of graphs; and the  $n$ -two-thirds upper bound conjecture (that is, Conjecture 2.3 in Chapter 2) by Foucaud and Henning [96] on the LTD-numbers of graphs — we recall the conjectures in the respective sections where they are studied.

Since the separation type considered in this chapter is clear, namely location or L-separation, we use the words “located” and “separated” interchangeably to mean that a pair of vertices of a graph are L-separated by some vertex subset of the graph. Also, recall that, given a vertex subset  $S$  of a graph  $G$ , an  $I$ -set of a vertex  $v \in V(G)$  with respect to  $S$  is given by the set  $I(v) = I_G(S; v) = N_G[v] \cap S$ . Thus, for  $v \in V(G) \setminus S$ , we have  $I_G(S; v) = N_G[v] \cap S = N_G(v) \cap S$ . Now, using the various definitions related to locating sets and Observations 2.3 and 2.4, we restate below (all in one place) the equivalent conditions to imply when a vertex subset of a graph  $G$  is a locating set of  $G$ .

**Remark 4.1.** *Let  $G$  be a graph and let  $S$  be a vertex subset of  $G$ . Then, the following assertions are equivalent.*

- (1)  $S$  is a locating set of  $G$ .
- (2) Each vertex in  $V(G) \setminus S$  has a unique neighborhood in  $S$ , that is, it has a unique  $I$ -set with respect to  $S$ .
- (3) For all distinct  $u, v \in V(G) \setminus S$ , we have  $N_G(u) \cap S \neq N_G(v) \cap S$ , that is,  $I_G(S; u) \neq I_G(S; v)$ .
- (4)  $S$  has non-empty intersection with  $\Delta_L(G; u, v) = \Delta_O(G; u, v) \cup \{u, v\} = \Delta_C(G; u, v) \cup \{u, v\}$  for all distinct  $u, v \in V(G)$ .
- (5)  $S$  has a non-empty intersection with
  - (a)  $N_G[u] \triangle N_G[v]$  for all pairs of non-adjacent vertices  $u, v \in V(G)$ , and
  - (b)  $N_G(u) \triangle N_G(v)$  for all pairs of adjacent vertices  $u, v \in V(G)$ .

The following remark (also shown in [11]) shows that in order to check if a dominating set (respectively, a total-dominating set)  $S$  of a graph  $G$  is an LD-code (respectively, an LTD-code) of  $G$ , we do not need to check if  $S$  separates every pair of distinct vertices in  $V(G) \setminus S$  but only those which are at a distance of at most 2.

**Remark 4.2.** *Let  $G$  be a graph. A dominating (respectively, a total-dominating) set  $C$  of  $G$  is an LD-code (respectively, an LTD-code) of  $G$  if and only if  $C$  separates every pair  $u, v$  of distinct vertices in  $V(G) \setminus S$  such that  $d_G(u, v) \leq 2$ .*

*Proof.* The necessary condition for the statement follows immediately from the definition of an LD-code and an LTD-code. We, therefore, prove the sufficient condition. Thus, it is enough to show

that  $C$  separates every pair  $u, v$  of distinct vertices in  $V(G) \setminus C$  such that  $d_G(u, v) \geq 3$ . So, assume  $u$  and  $v$  to be a pair of distinct vertices in  $V(G) \setminus C$  such that  $d_G(u, v) \geq 3$ . Since  $C$  is a dominating set of  $G$  and  $u \notin C$ , the vertex  $u$  must have a neighbor, say  $w$ , in  $C$ . However,  $w$  is not a neighbor of  $v$  (since  $d_G(u, v) \geq 3$ ) and hence,  $w$  must be a separating  $C$ -codeword of the pair  $u, v$  in  $G$ . This proves the result.  $\square$

## 4.1 Location with domination

In this section, we look at locating dominating codes of graphs from some selected graph families. In Section 4.1.2, we consider LD-codes of block graphs and study both upper and lower bounds on the LD-numbers of graphs of this class. Moreover, we consider the  $n$ -half upper bound Conjecture 2.2 by Garijo et al. [112] on the LD-numbers of block graphs and prove the conjecture to be true for such graphs. In addition, all bounds that we study in this section are tight with the LD-numbers of arbitrarily large-ordered block graphs attaining these bounds.

In Section 4.1.3, we again study Conjecture 2.2 for subcubic graphs and prove it to be true for graphs of this class. However, besides that, we also expand the scope of this conjecture by showing that it is also true for subcubic graphs even with closed twins and with open twins of degree 3 — except when the graph is either a  $K_3$  or a  $K_4$  or a  $K_{3,3}$ . Again, we show that this upper bound is tight for all cases for which it is true. The conjecture however fails for subcubic graphs with open twins of degree either 1 or 2 and we provide counterexamples to show this failure in such cases.

We recall here the  $n$ -half upper bound conjecture on LD-numbers of twin-free and isolate-free graphs by Garijo et al. in [112] with a slightly stronger reformulation by Foucaud and Henning in [96].

**Conjecture 2.2** ([112, 95]) *Let  $G$  be a twin-free and isolate-free graph on  $n$  vertices. Then, we have*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

Both restrictions in Conjecture 2.2 are required since every isolated vertex is required to be in a dominating set, the complete graph  $K_n$  (with closed twins) has  $\gamma^{\text{LD}}(K_n) = n - 1$  and the star  $K_{1,n-1}$  (with open twins) has  $\gamma^{\text{LD}}(K_{1,n-1}) = n - 1$ . Conjecture 2.2 has been studied in several other works like [31, 47, 65, 95, 97, 98, 112]. Also see [88] for a short introduction to this conjecture. Besides introducing the conjecture, Garijo et al. [112] gave a general upper bound  $\lfloor 2n/3 \rfloor + 1$  for the LD-number of twin-free graphs. Recently, Bousquet et al. [31] improved this general upper bound to  $\lfloor 5n/8 \rfloor$ . Besides proving general upper bounds, a lot of the research on this conjecture has concentrated on proving it for some graph families. In particular, Garijo et al. [112] proved the following useful theorem.

**Theorem 4.1** ([112]). *Let  $G$  be a connected twin-free graph without 4-cycles on  $n \geq 2$  vertices. Then, we have*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

Furthermore, Garijo et al. [112] proved the conjecture for graphs with independence number at least  $\lfloor n/2 \rfloor$  (in particular this family includes bipartite graphs) and graphs with clique number at least  $\lfloor n/2 \rfloor + 1$ . In [98], Foucaud et al. proved the conjecture for split and cobipartite graphs, in [47], Chakraborty et al. proved the conjecture for block graphs and in [97], Foucaud and Henning proved the conjecture for line graphs. The conjecture has also been proven for maximal outerplanar graphs in [65] by Claverol et al. and for cubic graphs in [95] by Foucaud and Henning. For a more schematic representation of the literature about Conjecture 2.2, we also refer the reader to Table 2.2.

Besides upper bounds, lower bounds for LD-codes have also been considered in the literature. In [187], Slater has given a lower bound for the location-domination number of  $r$ -regular graphs. In particular, the result states that  $\gamma^{\text{LD}}(G) \geq \frac{n}{3}$  for cubic graphs. However, the proof also holds for subcubic graphs in general.

### 4.1.1 Preliminary results

We now present some preliminary results with respect to LD-codes of graphs. Lemmas 4.1 and 4.2 have previously been considered for trees in [27] and [32]. The proofs for these generalizations follow similarly as the original ones but we offer them for the sake of completeness.

**Lemma 4.1.** *Let  $G$  be a connected graph on at least three vertices. Then  $G$  admits an optimal locating-dominating set  $S$  such that every support vertex of  $G$  is in  $S$ .*

*Proof.* Suppose to the contrary that there exists a graph  $G$  such that no optimal LD-code contains every support vertex of  $G$ . Furthermore, let  $S$  be an optimal LD-code of  $G$  such that it contains the largest number of support vertices among all optimal LD-codes of  $G$ . Furthermore, let  $s$  be a support vertex of  $G$  not in  $S$  and let  $u_1, u_2, \dots, u_k$  be all the leaves adjacent to  $s$ . Since  $S$  is a dominating set, we have  $u_i \in S$  for all  $i \in [k]$ . We show that, contrary to the maximality of  $S$ , the set  $S' = (S \setminus \{u_1\}) \cup \{s\}$  is an LD-code of  $G$ . Indeed, the set  $S \setminus \{u_1\}$  separates all pairs of vertices in  $V(G) \setminus (S \cup \{s, u_1\})$ . Furthermore, since  $S$  is a dominating set of  $G$  and  $s \notin S$ , any neighbor  $v$  other than  $u_1$  of  $s$  has  $(N_G[v] \setminus \{s\}) \cap S \neq \emptyset$ . This implies that the vertex  $u_1$  is the only vertex in  $V(G) \setminus S'$  with  $I$ -set  $\{s\}$  in  $S'$ . Therefore, set  $S'$  is an LD-code, a contradiction.  $\square$

**Lemma 4.2.** *Let  $G$  be a connected graph on at least three vertices without open twins of degree 1. Then  $G$  admits an optimal locating-dominating set  $S$  such that there are no leaves of  $G$  in  $S$ .*

*Proof.* Suppose to the contrary that there exists a graph  $G$  without open twins of degree 1 such that every optimal LD-code contains a leaf of  $G$ . Consider an optimal LD-code  $S$  such that it contains the least number of leaves among all optimal LD-codes of  $G$  which contains every support vertex of  $G$ . Notice that  $S$  exists by Lemma 4.1. Hence, there exist adjacent vertices  $s, u \in S$  such that  $s$  is a support vertex and  $u$  is a leaf. Since  $S$  is optimal, the set  $S \setminus \{u\}$  is not locating-dominating. Therefore, by the optimality of  $S$ , there exists a unique vertex  $v \notin S$  such that  $I(v) = \{s\}$ . Since  $G$  has no twins of degree 1, the vertex  $v$  is not a leaf. However, now the set  $S' = (S \cup \{v\}) \setminus \{u\}$  is an LD-code of  $G$ . Since  $S'$  is optimal, contains all support vertices and fewer leaves of  $G$  than  $S$ , it contradicts the minimality of  $S$ . Therefore, the claim follows.  $\square$

### 4.1.2 Block graphs

In this section, we prove Conjecture 2.2 for block graphs. We refer the reader to Section 2.1.3 for the definitions and notations concerning block graphs. We recall some of the essential notations here for the self-containment of the section. Let  $G$  be a connected block graph. Then,  $\mathcal{K}(G)$  denotes the set of all blocks of  $G$ . We fix a root block  $K_0 \in \mathcal{K}(G)$  and let  $f : \mathcal{K}(G) \rightarrow \mathbb{Z}$  be the layer function on  $G$ . For a block  $K$  of  $G$ , the notations  $art^-(K)$ ,  $art^+(K)$  and  $\overline{art}(K)$  denote the singleton set of negative articulation vertex, the set of all positive articulation vertices and that of all non-articulation vertices of  $K$ , respectively.

First, we prove two results on upper bounds for the LD-numbers of block graphs. The first result is a more general one in which the upper bound is in terms of the number of blocks and other quantities arising out of the structural properties of a block graph. On the other hand, the second result is proving Conjecture 2.2 for block graphs.

The results of this section have appeared in [47].

#### 4.1.2.1 Upper bounds on LD-numbers of block graphs

We begin with the more general result on LD-numbers of block graphs.

**Theorem 4.2.** *Let  $G$  be a connected block graph and  $\mathcal{K}(G)$  be the set of blocks in  $G$ . Then, we have*

$$\gamma^{\text{LD}}(G) \leq |\mathcal{K}(G)| + \sum_{\substack{K \in \mathcal{K}(G), \\ |\overline{art}(K)| \geq 2}} (|\overline{art}(K)| - 2).$$

*Proof.* We define a set  $C \subset V(G)$  by the following rules.

Rule 1: For every block  $K \in \mathcal{K}(G)$  which does not contain any closed-twins, that is, with at most one non-articulation vertex, pick an arbitrary vertex from  $V(K) \setminus \text{art}^-(K)$  in  $C$ .

Rule 2: For every block  $K \in \mathcal{K}(G)$  which contains closed-twins, that is, with at least two non-articulation vertices, pick any  $|\overline{\text{art}}(K)| - 1$  vertices from  $\overline{\text{art}}(K)$  in  $C$ .

Note that the vertices added in  $C$  by the above rules are all distinct. Therefore, the following is the cardinality of  $C$ .

$$\begin{aligned} |C| &= \left| \{K \in \mathcal{K}(G) : |\overline{\text{art}}(K)| \leq 1\} \right| + \sum_{\substack{K \in \mathcal{K}(G), \\ |\overline{\text{art}}(K)| \geq 2}} (|\overline{\text{art}}(K)| - 1) \\ &= |\mathcal{K}(G)| + \sum_{\substack{K \in \mathcal{K}(G), \\ |\overline{\text{art}}(K)| \geq 2}} (|\overline{\text{art}}(K)| - 2). \end{aligned}$$

The result, therefore, follows from proving that  $C$  is an LD-code of  $G$ .

First of all, we notice that, by the construction of  $C$ , for every block  $K \in \mathcal{K}(G)$ , there exists a vertex  $v_K \in V(K) \cap C$ . Therefore,  $C$  is a dominating set of  $G$ . We now show that  $C$  is also a locating set of  $G$ . So assume that  $u, v \in V(G) \setminus C$  are distinct vertices of  $G$ . Then, by the construction of  $C$ , we must have  $u \in V(K)$  and  $v \in V(K')$  for a distinct pair of blocks  $K, K' \in \mathcal{K}(G)$ .

■ **Claim 1.** There exist vertices  $v_K \in V(K) \cap C$  and  $v_{K'} \in V(K') \cap C$  such that  $v_K \neq v_{K'}$ .

*Proof of claim.* On the contrary, if  $V(K) \cap C = V(K') \cap C = \{v_K\}$ , this would imply that  $v_K$  is the negative articulation vertex of either  $K$  or  $K'$ . Without loss of generality, let us assume that  $\text{art}^-(K) = \{v_K\}$ . If  $K$  does not contain any closed twins, then by Rule 1, there exists a vertex in  $K$  other than  $v_K$  which belongs to  $C$ , a contradiction to  $V(K) \cap C = \{v_K\}$ . Therefore,  $K$  contains closed twins. In other words,  $|\overline{\text{art}}(K)| \geq 2$ . Now, by Rule 2, at least one vertex  $w$ , say, of  $\overline{\text{art}}(K)$  belongs to  $C$ . Since  $\overline{\text{art}}(K) \cap \text{art}^-(K) = \emptyset$ , we have  $w \neq v_K$  and thus, we run into the same contradiction. This proves the claim. ■

This implies that at least one of  $v_K$  and  $v_{K'}$  must open-separate  $u$  and  $v$  in  $G$  and, hence,  $C$  is a locating set of  $G$ . □

There is an infinite number of arbitrarily large connected block graphs whose LD-numbers attain the upper bound in Theorem 4.2. One such subfamily of block graphs is the following. For positive integers  $t \geq 2$  and  $m_1, m_2, \dots, m_t$  with  $m_i \geq 3$  for each  $i$ , we define a family of graphs  $S_t(m_1, m_2, \dots, m_t)$  by the following rule. Let  $X$  be a copy of the complete graph on  $t$  vertices and name its vertices  $v_1, v_2, \dots, v_t$ . Also, for all  $1 \leq i \leq t$ , let  $Y_i$  be a copy of the complete graph on  $m_i$  vertices. Let  $S_t(m_1, m_2, \dots, m_t)$  be the block graph obtained by identifying a vertex of  $Y_i$  with  $v_i$  of  $X$  for every  $1 \leq i \leq t$ . For brevity, we continue to call the identified vertices resulting in  $S_t(m_1, m_2, \dots, m_t)$  by the same names of  $v_1, v_2, \dots, v_t$  as before. See Figure 4.1 for an example of the graph  $S_t(m_1, m_2, \dots, m_t)$  constructed with  $t = 5$ , and  $m_1 = m_3 = 4$ ,  $m_2 = m_5 = 3$  and  $m_4 = 5$ . We then show the following.

**Proposition 4.1.** For  $t \geq 2$ , and  $m_1, m_2, \dots, m_t$  such that  $m_i \geq 3$  for all  $1 \leq i \leq t$ , we have

$$\gamma^{\text{LD}}(S_t(m_1, m_2, \dots, m_t)) = |\mathcal{K}(S_t(m_1, m_2, \dots, m_t))| + \sum_{\substack{K \in \mathcal{K}(S_t(m_1, m_2, \dots, m_t)), \\ |\overline{\text{art}}(K)| \geq 2}} (|\overline{\text{art}}(K)| - 2).$$

*Proof.* Let  $G = S_t(m_1, m_2, \dots, m_t)$ . We note here that the number of blocks in  $G$  is  $t + 1$ ; and the only blocks  $K \in \mathcal{K}(G)$  with  $|\overline{\text{art}}(K)| \geq 2$  are  $Y_1, Y_2, \dots, Y_t$  (as per the notations used in the

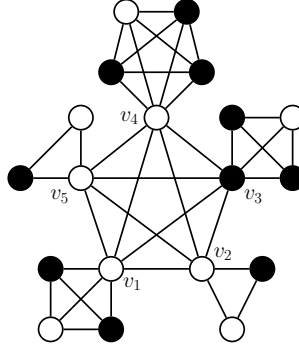


Figure 4.1: Graph  $S_5(4, 3, 4, 5, 3)$  whose LD-number attains the upper bound in Theorem 4.2. The black vertices represent those included in the LD-code  $C$  of  $G$  as described in the proof of Theorem 4.2.

preceding discussion). More precisely, for each  $1 \leq i \leq t$ , we have  $|\overline{art}(Y_i)| = m_i - 1$ . So, the upper bound for the LD-number of  $G$  by Theorem 4.2 is  $t + 1 + \sum_{i=1}^t (m_i - 3) = 1 - 2t + \sum_{i=1}^t m_i$ . Now, assume that  $C$  is a minimum LD-code of  $G$ . Let us first assume that  $V(X) \cap C = \emptyset$ . Then, since any two vertices in  $V(Y_i)$  have the same neighborhood in  $C$ , it implies that we must have  $|V(Y_i) \cap C| = m_i - 1$ . This further implies that  $|C| = \sum_{i=1}^t (m_i - 1) = -t + \sum_{i=1}^t m_i > 1 - 2t + \sum_{i=1}^t m_i$  (since  $t \geq 2$ ), the upper bound by Theorem 4.2 resulting in a contradiction. Therefore, we must have  $V(X) \cap C \neq \emptyset$ . Thus, let  $v_i \in C$  for some  $1 \leq i \leq t$ . Note that  $v_i \in V(Y_i)$ . Since any two vertices of  $V(Y_i) \setminus \{v_i\}$  are twins in  $G$ , we have  $|V(Y_i) \cap C| \geq m_i - 1$ . Moreover, for  $1 \leq j \leq t$  such that  $j \neq i$ , again, since any two vertices of  $V(Y_j) \setminus \{v_j\}$  are twins in  $G$ , we now have  $|V(Y_j) \cap C| \geq m_j - 2$ . Hence, we have  $|C| \geq |V(Y_i) \cap C| + \sum_{j=1, j \neq i}^t |V(Y_j) \cap C| \geq 1 - 2t + \sum_{i=1}^t m_i$ .  $\square$

We now prove Conjecture 2.2 for twin-free block graphs.

**Theorem 4.3.** *Let  $G$  be a connected twin-free block graph on  $n$  vertices. Then  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$ .*

*Proof.* To prove the theorem, we partition the vertex set of  $G$  into two special subsets  $C^*$  and  $D^*$ .

Assume that  $K_0 \in \mathcal{K}(G)$  is a leaf block of  $G$ . Then,  $|V(K_0)| = 2$ , as  $G$  is twin-free. Assign  $K_0$  to be the root block of  $G$ , that is, define a layer function  $f : \mathcal{K}(G) \rightarrow \mathbb{Z}$  on  $G$  such that  $f(K_0) = 0$ . We then construct the sets  $C^*$  and  $D^*$  by the following rules applied inductively on  $i \in f(\mathcal{K}(G))$ . See Figure 4.2 for a demonstration of this construction.

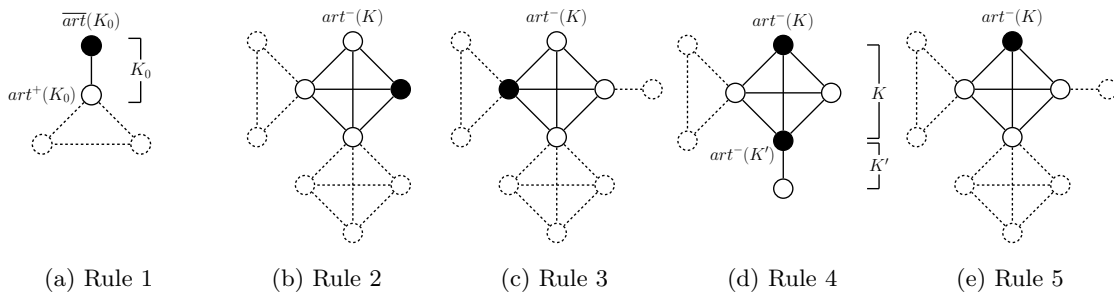


Figure 4.2: Example of each rule in the proof of Theorem 4.3 for the construction of the sets  $C^*$  and  $D^*$ . In each example, the black vertices represent those picked in  $C^*$  and the white vertices represent those picked in  $D^*$ . The blocks with solid edges represent those blocks (in the  $i$ -th layer, say) from which vertices are chosen either in  $C^*$  or  $D^*$ . The blocks with dashed edges represent those blocks in the next layer (the  $(i + 1)$ -th) which, inductively, are yet to be analyzed for their choices of vertices in  $C^*$  and  $D^*$ ; but whose presence in the figure is necessary to determine the positive, negative and the non-articulation vertices of the block in the  $i$ -th layer.

- Rule 1: Pick the (positive) articulation vertex of the root block  $K_0$  in  $D^*$  (that is, let  $\text{art}^+(K_0) \subseteq D^*$ ) and pick the (other) non-articulation vertex of  $K_0$  in  $C^*$  (that is, let  $\overline{\text{art}}(K_0) \subseteq C^*$ ). See Figure 4.2a for an example.
- Rule 2: For every non-root block  $K \in \mathcal{K}(G)$  with at least one non-articulation vertex (that is,  $\overline{\text{art}}(K) \neq \emptyset$ ) and whose negative articulation vertex is in  $D^*$  (that is,  $\text{art}^-(K) \subseteq D^*$ ), pick all non-articulation vertices of  $K$  in  $C^*$  (that is, let  $\overline{\text{art}}(K) \subseteq C^*$ ); and all positive articulation vertices of  $K$  in  $D^*$  (that is, let  $\text{art}^+(K) \subseteq D^*$ ). See Figure 4.2b for an example.
- Rule 3: For every non-root block  $K \in \mathcal{K}(G)$  with no non-articulation vertices (that is,  $\overline{\text{art}}(K) = \emptyset$ ) and whose negative articulation vertex is in  $D^*$  (that is,  $\text{art}^-(K) \subseteq D^*$ ), pick *one* positive articulation vertex, say,  $w$  of  $K$  in  $C^*$  and the rest of the positive articulation vertices in  $D^*$  (that is, let  $\text{art}^+(K) \setminus \{w\} \subseteq D^*$ ). See Figure 4.2c for one such case.
- Rule 4: For every non-root block  $K \in \mathcal{K}(G)$  with at least one non-articulation vertex (i.e.  $\overline{\text{art}}(K) \neq \emptyset$ ) and whose negative articulation vertex is in  $C^*$  (i.e.  $\text{art}^-(K) \subseteq C^*$ ), pick *one* positive articulation vertex (if available), say  $w$ , of  $K$  in  $C^*$ ; and pick all other vertices in  $V(K)$ , except the vertex  $w$  and the negative articulation vertex of  $K$ , in  $D^*$  (i.e. let  $V(K) \setminus (\text{art}^-(K) \cup \{w\}) \subseteq D^*$ ). See Figure 4.2d for both examples of when articulation vertices are available (block  $K$ ) and when they are not which is in the case of leaf blocks (block  $K'$ ).
- Rule 5: For every non-root block  $K \in \mathcal{K}(G)$  with no non-articulation vertices (that is,  $\overline{\text{art}}(K) = \emptyset$ ) and whose negative articulation vertex is in  $C^*$  (i.e.  $\text{art}^-(K) \subseteq C^*$ ), pick all positive articulation vertices of  $K$  in  $D^*$  (i.e.  $\text{art}^+(K) \subseteq D^*$ ). See Figure 4.2e for an illustration.

From the construction,  $C^*$  and  $D^*$  are complements of each other in  $V(G)$ . We claim that both  $C^*$  and  $D^*$  are LD-codes of  $G$ . We first show that both are dominating sets of  $G$ .

■ **Claim 2.** Both  $C^*$  and  $D^*$  are dominating sets of  $G$ .

*Proof of claim.* To prove that both  $C^*$  and  $D^*$  are dominating sets of  $G$ , it is enough to show that, for every block  $K \in \mathcal{K}(G)$ , both  $V(K) \cap C^* \neq \emptyset$  and  $V(K) \cap D^* \neq \emptyset$ . By Rule 1, the claim is true for the root block  $K_0$ . So, assume  $K \in \mathcal{K}(G)$  to be a non-root block. First, suppose that the negative articulation vertex of  $K$  belongs to  $D^*$ . Then, by Rules 2 and 3, we have  $V(K) \cap C^* \neq \emptyset$ . Next, suppose that the negative articulation vertex of  $K$  belongs to  $C^*$ . Then, by Rules 4 and 5, we have  $V(K) \cap D^* \neq \emptyset$ . ■

We now show that both  $C^*$  and  $D^*$  are also locating sets of  $G$ . We start with  $C^*$ .

■ **Claim 3.**  $C^*$  is a locating set of  $G$ .

*Proof of claim.* Assume that  $u, v \in D^*$  are distinct vertices of  $G$ . Since  $G$  is twin-free, there exist distinct blocks  $K, K' \in \mathcal{K}(G)$  such that  $u \in V(K)$  and  $v \in V(K')$ . By the proof of Claim 2, there exist vertices  $v_K \in V(K) \cap C^*$  and  $v_{K'} \in V(K') \cap C^*$ . If  $v_K \neq v_{K'}$ , then either one of  $v_K$  and  $v_{K'}$  must locate  $u$  and  $v$  in  $G$ . So, let us assume that no such pairs of distinct vertices  $v_K \in V(K) \cap C^*$  and  $v_{K'} \in V(K') \cap C^*$  exist, that is,  $V(K) \cap V(K') \subseteq C^*$  and that  $V(K) \Delta V(K') \subseteq D^*$ .

We now claim that either  $u$  is an articulation vertex of  $K$  or  $v$  is an articulation vertex of  $K'$  (or both). So, toward a contradiction, suppose that both  $u$  and  $v$  are non-articulation vertices of  $K$  and  $K'$ , respectively. Then the following two cases arise.

► **Case 1:  $K$  and  $K'$  belong to different layers.**

Without loss of generality, assume that  $f(K') = f(K) + 1$ . Then,  $K \neq K_0$ , or else, by Rule 1,  $u$ , being a non-articulation vertex of  $K$ , must belong to  $C^*$ , contrary to our assumption. Therefore,  $K$  is a non-leaf block. Now,  $V(K) \Delta V(K') \subseteq D^*$  implies that the negative articulation vertex of  $K$  belongs to  $D^*$ . Since  $u$  is a non-articulation vertex of  $K$ , by Rule 2,  $u \in C^*$  which is a contradiction to our assumption.

► **Case 2:  $K$  and  $K'$  belong to the same layer.**

In this case,  $K$  and  $K'$  cannot both be leaf blocks, or else,  $G$  would have twins. So, without loss of generality, suppose that  $K$  is a non-leaf block. Now, the negative articulation vertex of  $K$  belongs to  $C^*$ . Since  $K$  is a non-leaf block, there exists a positive articulation vertex of  $K$  and, hence, by Rule 4,  $art^+(K) \cap C^* \neq \emptyset$  which contradicts the fact that  $V(K) \triangle V(K') \subseteq D^*$ .

This proves our claim that either  $u$  is an articulation vertex of  $K$  or  $v$  is an articulation vertex of  $K'$  (or both). So if, without loss of generality, we assume that  $u$  is an articulation vertex of  $K$ , then  $\{u\} = V(K) \cap V(K'')$  for some block  $K'' (\neq K) \in \mathcal{K}(G)$ . Moreover,  $K'' \neq K'$ , or else,  $V(K) \cap V(K') = \{u\} \subseteq D^*$  which contradicts our assumption that  $V(K) \cap V(K') \subseteq C^*$ . Hence, some vertex in  $V(K'') \cap C^*$  (which exists due to the proof of Claim 2) must open-separate  $u$  and  $v$  in  $G$ . This proves our current claim. ■

We now prove the same for  $D^*$ .

■ **Claim 4.**  $D^*$  is a locating set of  $G$ .

*Proof of claim.* Assume that  $u, v \in C^*$  are distinct vertices of  $G$ . Since  $G$  is twin-free, there exist distinct blocks  $K, K' \in \mathcal{K}(G)$  such that  $u \in V(K)$  and  $v \in V(K')$ . By the proof of Claim 2, there exist vertices  $v_K \in V(K) \cap D^*$  and  $v_{K'} \in V(K') \cap D^*$ . If  $v_K \neq v_{K'}$ , then either one of  $v_K$  and  $v_{K'}$  must open-separate  $u$  and  $v$  in  $G$ . So, let us assume that no such pairs of distinct vertices  $v_K \in V(K) \cap D^*$  and  $v_{K'} \in V(K') \cap D^*$  exist, that is,  $V(K) \cap V(K') \subseteq D^*$  and that  $V(K) \triangle V(K') \subseteq C^*$ .

We now claim that either  $u$  is an articulation vertex of  $K$  or  $v$  is an articulation vertex of  $K'$  (or both). So, toward a contradiction, suppose that both  $u$  and  $v$  are non-articulation vertices of  $K$  and  $K'$ , respectively. Then the following two cases arise.

► **Case 1:  $K$  and  $K'$  belong to different layers.**

Without loss of generality, assume that  $f(K') = f(K) + 1$ . If  $|V(K')| \geq 3$ , since  $G$  is twin-free and since  $v$  is a non-articulation vertex of  $K'$ , then  $K'$  contains exactly one non-articulation vertex and thus,  $art^+(K') \cap D^* \neq \emptyset$  by Rule 2. This, however, is a contradiction to the fact that  $V(K) \triangle V(K') \subseteq C^*$ . So, assume that  $|V(K')| = 2$ , in which case,  $K'$  is a leaf block (since, again,  $v$  is a non-articulation vertex of  $K'$ ). This implies that  $K$  is a non-leaf block, or else,  $G$  would have twins. So, in particular,  $K \neq K_0$ , the root block of  $G$ . Moreover,  $V(K) \triangle V(K') \subseteq C^*$  implies that the negative articulation vertex of  $K$  belongs to  $C^*$ . Therefore, since  $u$  is a non-articulation vertex of  $K$ , by Rule 4,  $u \in D^*$  which is a contradiction to our assumption.

► **Case 2:  $K$  and  $K'$  belong to the same layer.**

In this case,  $K$  and  $K'$  cannot both be leaf blocks, or else,  $G$  would have twins. So, without loss of generality, assume  $K$  to be a non-leaf block. Therefore,  $|V(K)| \geq 3$ , or else,  $u$  would be an articulation vertex of  $K$ , contrary to our assumption. The negative articulation vertex of  $K$  belongs to  $D^*$ . Therefore, by Rule 2,  $art^+(K) \cap D^* \neq \emptyset$  which contradicts  $V(K) \triangle V(K') \subseteq C^*$ .

This, therefore, proves our claim that either  $u$  is an articulation vertex of  $K$  or  $v$  is an articulation vertex of  $K'$  (or both). If, without loss of generality,  $u$  is an articulation vertex of  $K$ , then  $\{u\} = V(K) \cap V(K'')$  for some block  $K'' (\neq K) \in \mathcal{K}(G)$ . Moreover,  $K'' \neq K'$ , or else,  $V(K) \cap V(K') = \{u\} \subseteq C^*$  which contradicts our assumption that  $V(K) \cap V(K') \subseteq D^*$ . Hence, some vertex in  $V(K'') \cap D^*$  (which exists due to the proof of Claim 2) must open-separate  $u$  and  $v$  in  $G$ . This, again, proves our current claim. ■

Combining Claims 2, 3 and 4, we find that  $C^*$  and  $D^*$  are both LD-codes of the twin-free block graph  $G$  with no isolated vertices. Moreover, since  $C^*$  and  $D^*$  are complements of each other in  $V(G)$ , at least one of them must have cardinality of at most half the order of  $G$ . This proves the theorem. □



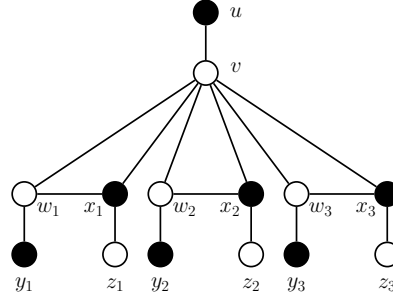


Figure 4.3: Graph  $H_3$  whose LD-number attains the upper bound in Theorem 4.3. The black vertices represent those included in the LD-code  $C^*$  of  $G$  described in the proof of Theorem 4.3.

Theorem 4.3 therefore proves Conjecture 2.2 for block graphs.

**Corollary 4.1.** *Let  $G$  be a twin-free and isolate-free block graph on  $n$  vertices, then  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$ .*

The trees attaining the bound of Theorem 4.3 were characterized in [96]. There are also arbitrarily large twin-free block graphs that are not trees and whose LD-numbers attain the bound given in Theorem 4.3. To demonstrate this attainment, we look at the following subfamily of block graphs which we denote by  $H_t$ : For a fixed integer  $t \geq 1$ , let  $T_1, T_2, \dots, T_t$  be  $t$  copies of  $K_3$ , the complete graph on three vertices. Suppose that  $V(T_i) = \{v_i, w_i, x_i\}$  for each  $1 \leq i \leq t$ . Also, let  $R, R_1, R_2, \dots, R_t, R'_1, R'_2, \dots, R'_t$  be  $2t + 1$  copies of  $P_2$ , the path on two vertices. Also, let  $V(R) = \{u, v\}$  and for all  $1 \leq i, i' \leq t$ , let  $V(R_i) = \{y'_i, y_i\}$  and  $V(R'_{i'}) = \{z'_{i'}, z_{i'}\}$ . We then identify the vertices  $v, v_1, v_2, \dots, v_t$  to a single vertex which we continue to call  $v$ ; and, for each  $1 \leq i \leq t$ , we identify the vertices  $w_i$  and  $y'_i$  to a single vertex and the vertices  $x_i$  and  $z'_{i'}$  to a single vertex. In the latter two cases, we continue to call the identified vertices  $w_i$  and  $x_i$ , respectively. The new resulting graph is what we call  $H_t$ . See Figure 4.3 for an example of  $H_t$  with  $t = 3$ . With that, we now prove the following.

**Proposition 4.2.** *For each integer  $t \geq 1$ ,  $\gamma^{\text{LD}}(H_t) = \frac{|V(H_t)|}{2}$ .*

*Proof.* Notice that the graph  $H_t$  is twin-free. Since  $|V(H_t)| = 4t + 2$ , we therefore have from Theorem 4.3 that  $\gamma^{\text{LD}}(H_t) \leq 2t + 1$ .

We now prove that  $\gamma^{\text{LD}}(H_t) \geq 2t + 1$ . Since each of the  $2t + 1$  edges  $uv, w_i y_i, x_i z_i$  (for  $1 \leq i \leq t$ ) of  $H_t$  contains a vertex of degree 1, therefore any LD-code of  $H_t$ , by its property of domination, must contain at least one endpoint of each of these edges. Since the above edges are all pairwise disjoint, any LD-code of  $H_t$  must contain at least  $2t + 1$  vertices of  $H_t$ .  $\square$

#### 4.1.2.2 Lower bounds on LD-numbers of block graphs

We now turn to studying lower bounds on the LD-numbers of block graphs. We find the lower bounds of LD-numbers both in terms of the number of vertices and in terms of the number of blocks of a block graph. For the rest of this section, given a block graph  $G$ , by  $\mathcal{K}_{\text{leaf}}(G)$  we denote the set of all leaf blocks of  $G$  with at least one edge in the block. Moreover, by the symbol  $n_i(G)$ , we denote the number of vertices of degree  $i$  in the graph  $G$ .

##### Lower bound in terms of the order of the graph.

With respect to the order of a graph, we prove the following tight upper bound of the LD-number of a block graph.

**Theorem 4.4.** *Let  $G$  be a connected block graph. Then we have*

$$\gamma^{\text{LD}}(G) \geq \frac{|V(G)| + 1}{3}.$$

To prove the theorem, we first establish some lemmas and introduce some definitions.

**Lemma 4.3.** *Let  $G$  be a connected block graph with at least one edge and with blocks  $B_1, B_2, \dots, B_h$ , say. Then, there exist distinct vertices  $v_0, v_1, v_2, \dots, v_h$  of  $G$  such that  $v_0, v_1 \in V(B_1)$  and  $v_i \in V(B_i)$  for all  $2 \leq i \leq h$ .*

*Proof.* Since  $G$  has at least one edge, note that every block of  $G$  has at least one edge. The proof is by induction on  $h$  with the base case being  $h = 1$ . In the base case, we have  $G = B_1$ . Since  $G$  has at least one edge  $v_0v_1$ , say, the two distinct vertices are  $v_0$  and  $v_1$  and we are done. So, let us assume the induction hypothesis for any connected block graph  $G'$  with at least one edge and with  $h'$  blocks, where  $1 \leq h' \leq h - 1$ . Therefore, let  $h \geq 2$ . Without loss of generality, let us assume that  $B_h$  is a leaf block of  $G$ . Let  $G' = G - (V(B_h) \setminus \text{art}^-(B_h))$ . Then  $G'$  is also a connected block graph on  $h - 1$  blocks. Since  $G'$  contains the block  $B_1$ , the former has at least one edge. Then by the induction hypothesis, there exist distinct vertices  $v_0, v_1, v_2, \dots, v_{h-1}$  of  $G$  such that  $v_0, v_1 \in V(B_1)$  and  $v_i \in V(B_i)$  for all  $2 \leq i \leq h - 1$ . Since  $B_h$  has at least one edge, it implies that there exists a vertex  $v_h \in V(B_h) \setminus \text{art}^-(B_h)$ . This implies that  $v_h \neq v_i$  for all  $1 \leq i \leq h - 1$  and hence, the result holds.  $\square$

**Lemma 4.4.** *Let  $G$  be a connected block graph with at least one edge. Then we have*

$$|\mathcal{K}(G)| \leq |V(G)| - 1 - |\mathcal{K}_{\text{leaf}}(G)| + n_1(G).$$

*Proof.* Let  $\mathcal{L}(G) = \{L \in \mathcal{K}_{\text{leaf}}(G) : L \cong K_2\}$  and  $G^*$  be a graph obtained from  $G$  by, for each  $L \in \mathcal{L}(G)$ , introducing a new vertex and making it adjacent to both elements of  $V(L)$ . Thus,  $G^*$  is a block graph in which every leaf block has at least three vertices. We also note here that

- (1)  $|\mathcal{L}(G)| = n_1(G)$ ,
- (2)  $|V(G^*)| = |V(G)| + |\mathcal{L}(G)| = |V(G)| + n_1(G)$ ,
- (3)  $|\mathcal{K}(G)| = |\mathcal{K}(G^*)|$  and that
- (4)  $|\mathcal{K}_{\text{leaf}}(G)| = |\mathcal{K}_{\text{leaf}}(G^*)|$ .

Now, let  $|\mathcal{K}(G^*)| = h$  and  $\mathcal{K}(G^*) = \{B_1, B_2, \dots, B_h\}$ . Then, by Lemma 4.3, there exist distinct vertices  $v_0, v_1, v_2, \dots, v_h$  of  $G$  such that  $v_0, v_1 \in V(B_1)$  and  $v_i \in V(B_i)$  for all  $2 \leq i \leq h$ . Moreover, since at least one vertex in each leaf block of  $G^*$  is not any of the vertices  $v_i$ , we have

$$\begin{aligned} |\mathcal{K}(G)| &= |\mathcal{K}(G^*)| = h = |\{v_0, v_1, v_3, \dots, v_h\}| - 1 \leq |V(G^*)| - |\mathcal{K}_{\text{leaf}}(G^*)| - 1 \\ &= |V(G)| + n_1(G) - |\mathcal{K}_{\text{leaf}}(G)| - 1. \end{aligned} \quad \square$$

**Corollary 4.2.** *Let  $G$  be a block graph with  $k$  components. Then, we have*

$$|\mathcal{K}(G)| - n_0(G) \leq |V(G)| - k - |\mathcal{K}_{\text{leaf}}(G)| + n_1(G).$$

*Proof.* Assume that  $k = p + q$  such that  $G_1, G_2, \dots, G_p$  are the components of  $G$ , each with at least one edge; and that  $S_1, S_2, \dots, S_q$  are the components of  $G$ , each with a single vertex. Then, we have

$$\begin{aligned} |\mathcal{K}(G)| &= \sum_{1 \leq i \leq p} |\mathcal{K}(G_i)| + \sum_{1 \leq j \leq q} |\mathcal{K}(S_j)| \\ &\leq q - p + \sum_{1 \leq i \leq p} (|V(G_i)| - |\mathcal{K}_{\text{leaf}}(G_i)| + n_1(G_i)) \quad [\text{using Lemma 4.4}] \\ &= |V(G)| - k - |\mathcal{K}_{\text{leaf}}(G)| + n_1(G) + n_0(G) \quad [\text{since } q = n_0(G)] \end{aligned} \quad \square$$

**Corollary 4.3.** *Let  $G$  be a block graph with  $k$  components. Then we have*

$$|\mathcal{K}(G)| - n_0(G) \leq |V(G)| - k.$$

*Proof.* The result follows from Corollary 4.2 and the fact that  $n_1(G) \leq |\mathcal{K}_{\text{leaf}}(G)|$ .  $\square$

Before we come to our results, we define the following notations.

**Definition 4.1.** For a given code  $C$  (not necessarily an LD-code) of a connected block graph  $G$ , let us assume that the sets  $C_1, C_2, \dots, C_k$  partition the code  $C$  such that the induced subgraphs  $G[C_1], G[C_2], \dots, G[C_k]$  of  $G$  are the  $k$  components of the subgraph  $G[C]$  of  $G$  induced by  $C$ . Note that each  $C_i$  is a block graph (since every induced subgraph of a block graph is also a block graph). Then, the vertex set  $V(G)$  is partitioned into the four following parts.

- (1)  $V_1 = C$ ,
- (2)  $V_2 = \{v \in V(G) \setminus V_1 : |N_G(v) \cap C| = 1\}$ ,
- (3)  $V_3 = \{v \in V(G) \setminus V_1 : \text{there exist distinct } i, j \leq k \text{ such that } N_G(v) \cap C_i \neq \emptyset \text{ and } N_G(v) \cap C_j \neq \emptyset\}$ ,
- (4)  $V_4 = V(G) \setminus (V_1 \cup V_2 \cup V_3)$ . Note that, for all  $v \in V_4$ , we have  $N_G(v) \cap C \subset C_i$  for some  $i$  and that  $|N_G(v) \cap C_i| \geq 2$ .

**Definition 4.2.** Given a connected block graph  $G$  and a code  $C$  (not necessarily an LD-code) of  $G$ , let  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Moreover, let the vertex set  $V(G)$  be partitioned into the subsets  $V_1, V_2, V_3, V_4$  as in Definition 4.1. Then, consider the bipartite graph  $F_C(G)$ , where  $A = \{a_j : v_j \in V_3\}$  and  $B = \{u_i : G[C_i] \text{ is a component of } G[C]\}$  are the two parts of  $V(F_C(G))$ . As for the edge set  $E(F_C(G))$ , for each vertex  $v_j$  in  $V_3$ , we add an edge between  $a_j$  and  $u_i$  if  $v_j$  is adjacent to a vertex in  $C_i$ . The graph  $F_C(G)$  is called the *auxiliary graph* of  $G$ .

**Lemma 4.5.** For a connected block graph  $G$  and a code  $C$  (not necessarily an LD-code) of  $G$ , the auxiliary graph  $F_C(G)$  is a forest.

*Proof.* If there is a cycle in  $F_C(G)$ , there would be a cycle in  $G$  involving two vertices of different components  $G[C_i]$  and  $G[C_j]$ , say. By the definition of a block graph, the latter cycle in  $G$  has to induce a complete subgraph in  $G$ . However, that would imply that  $G[C_i]$  and  $G[C_j]$  must be the same component of  $G[C]$  which is a contradiction. Thus,  $F_C(G)$  is cycle-free and, hence, is a forest.  $\square$

We now prove a series of lemmas establishing upper bounds on the orders of each of the vertex subsets  $V_2, V_3$  and  $V_4$  of a connected block graph  $G$ .

**Lemma 4.6.** Let  $G$  be a connected block graph and  $C$  be an LD-code of  $G$ . Then, we have  $|V_2| \leq |C|$ .

*Proof.* By definition of  $V_2$ , each vertex  $v \in V_2$  has a unique neighbor  $u$  in  $C$ , that is,  $N_G(v) \cap C = \{u\}$ . Hence, there can be at most  $|C|$  vertices in  $V_2$ .  $\square$

**Lemma 4.7.** Let  $G$  be a connected block graph,  $C$  be an code (not necessarily an LD-code) of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Then, we have  $|V_3| \leq k - 1$ .

*Proof.* By Lemma 4.5,  $F_C(G)$  is a forest. Let  $V(F_C(G)) = A \sqcup B$  be as defined above. Then we have  $|B| = k$ . Therefore,  $F_C(G)$ , on account of being a forest, has at most  $|A| + k - 1$  edges. Also, since  $F_C(G)$  is bipartite, its number of edges is  $\sum_{a \in A} \deg_{F_C(G)}(a) \geq 2|A|$  (the last inequality holds since any vertex in the part  $A$  of  $V(F_C(G))$  is adjacent to at least two distinct vertices of  $B$ ). Thus, the result follows from the fact that  $|V_3| = |A|$ .  $\square$

**Lemma 4.8.** Let  $G$  be a connected block graph,  $C$  be an LD-code of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Then, we have  $|V_4| \leq |C| - k$ .

*Proof.* Let  $v$  be any vertex in  $V_4$  and let  $G[C_i]$  be the component of  $G[C]$  such that  $N_G(v) \cap C \subseteq C_i$ . Moreover,  $|N_G(v) \cap C_i| \geq 2$ . Then, notice that  $N_G(v) \cap C$  must be a subset of exactly one block of  $G[C_i]$ , or else,  $G[C_i]$  would be disconnected, as  $v \notin C$ . This implies that  $|V_4| \leq |\mathcal{K}(G[C])| - n_0(G[C]) \leq |C| - k$ , by Corollary 4.3. This proves the lemma.  $\square$

This brings us to the proof of Theorem 4.4.

**Theorem 4.4.** *Let  $G$  be a connected block graph. Then we have*

$$\gamma^{\text{LD}}(G) \geq \frac{|V(G)| + 1}{3}.$$

*Proof.* Let  $|V(G)| = n$ . Assume  $C$  to be an LD-code of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Recalling from Definition 4.1 the sets  $V_1, V_2, V_3, V_4$  that partition  $V(G)$ , we prove the theorem using the relation  $|V(G)| = |C| + |V_2| + |V_3| + |V_4|$  and the upper bounds for  $|V_2|$ ,  $|V_3|$  and  $|V_4|$  in Lemmas 4.6, 4.7 and 4.8, respectively. Therefore, we have

$$\begin{aligned} n &= |C| + |V_2| + |V_3| + |V_4| \\ &\leq |C| + |C| + k - 1 + |C| - k \\ &= 3|C| - 1 \end{aligned}$$

and, hence, the result holds.  $\square$

We now look at examples of connected block graphs that are extremal with respect to Theorem 4.4. To that end, we show that there are infinitely many connected block graphs whose LD-numbers reach the bound of Theorem 4.4.

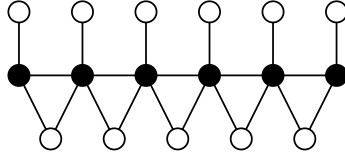


Figure 4.4: Extremal case where the lower bound in Theorem 4.4 is attained. The black vertices form a minimum LD-code.

**Proposition 4.3.** *There exist arbitrarily large connected block graphs whose LD-number attains the lower bound in Theorem 4.4.*

*Proof.* For any  $\ell \geq 1$ , consider a path on vertices  $u_1, u_2, \dots, u_\ell$ . For all  $1 \leq i \leq \ell$ , attach a vertex  $v_i$  by the edge  $v_i u_i$  and, for each pair  $u_i, u_{i+1}$  for  $\ell \geq 2$  and  $1 \leq i \leq \ell - 1$ , attach a vertex  $w_i$  by edges  $w_i u_i$  and  $w_i u_{i+1}$ . We call the graph  $G$ . See Figure 4.4 for an example with  $k = 6$ . Then it can be verified that  $C = \{u_1, u_2, \dots, u_\ell\}$  is a minimum LD-code of  $G$ . Since  $|V(G)| = 3\ell - 1$ , the LD-number of  $G$  attains the lower bound.  $\square$

### Lower bound in terms of the number of blocks

Consider the parameter  $|\mathcal{K}(G)|$ , we can use the relation  $|V(G)| \geq |\mathcal{K}(G)| + 1$  to obtain a similar lower bound. But this lower bound can be improved as the next theorem shows.

**Theorem 4.5.** *Let  $G$  be a connected block graph and  $\mathcal{K}(G)$  be the set of all blocks of  $G$ . Then we have*

$$\gamma^{\text{LD}}(G) \geq \frac{|\mathcal{K}(G)| + 2}{3}.$$

Again, to prove the above theorem, we first establish a series of lemmas.

**Definition 4.3.** Let  $C$  be a code (not necessarily an LD-code) of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . First, we define  $\mathcal{I}_G(C) = \{K \in \mathcal{K}(G) : V(L) \subset V(K) \text{ for some } L \in \mathcal{K}(G[C])\}$ . Moreover, for each  $1 \leq i \leq k$ , let  $\mathcal{I}_G(C_i) = \{K \in \mathcal{K}(G) : V(L) \subset V(K) \text{ for some } L \in \mathcal{K}(G[C_i])\}$ . Next, we define the following types of blocks of  $G$ .

- (1) Let  $\mathcal{K}_C(G) = \{K \in \mathcal{I}_G(C) : V(K) \subset C\}$ , that is, all blocks of  $G$  which are also blocks of the subgraph  $G[C]$  (also a block graph) of  $G$ .
- (2) Let  $\mathcal{K}_{\overline{C}}(G) = \mathcal{K}(G) \setminus \mathcal{I}_G(C)$ . In other words, the set  $\mathcal{K}_{\overline{C}}(G)$  includes all blocks of  $G$  which do not contain any vertices of the code  $C$ .
- (3) For  $i = 2, 3, 4$ , let  $\mathcal{K}_i(G) = \{K \in \mathcal{I}_G(C) : V(K) \cap V_i \neq \emptyset\}$  (recall the sets  $V_1, V_2, V_3, V_4$  from Definition 4.1).

We note here that,  $\mathcal{K}(G) = \mathcal{K}_C(G) \cup \mathcal{K}_{\overline{C}}(G) \cup \mathcal{K}_2(G) \cup \mathcal{K}_3(G) \cup \mathcal{K}_4(G)$ . We now have the following bounds.

**Lemma 4.9.**  $|\mathcal{K}_2(G)| \leq |V_2|$ .

*Proof.* Since each vertex in the set  $V_2$  belongs to a unique block  $K \in \mathcal{K}_2(G)$ , this claim is true.  $\square$

We now invoke the auxiliary graph  $F_C(G)$  of  $G$  from Definition 4.2 and assume that there are  $l$  components of  $F_C(G)$ . Then we have the following claim.

**Lemma 4.10.**  $|\mathcal{K}_3(G)| \leq 2(k - l)$ .

*Proof.* Since each vertex of  $F_C(G)$  in the part  $A$  is of degree at least 2, we have  $|E(F_C(G))| \geq 2|A| = 2|V_3|$ . Combining this with the fact that  $|E(F_C(G))| = |V_3| + k - l$  (since  $F_C(G)$  is a forest by Lemma 4.5), we have  $|V_3| \leq k - l$ . Hence, we have  $|\mathcal{K}_3(G)| \leq |E(F_C(G))| \leq 2(k - l)$ .  $\square$

**Lemma 4.11.**  $|\mathcal{K}_{\overline{C}}(G)| \leq l - 1$ .

*Proof.* Let  $F_1, F_2, \dots, F_l$  be the  $l$  components of the auxiliary (bipartite and forest) graph  $F_C(G)$  of  $G$ . To count  $|\mathcal{K}_{\overline{C}}(G)|$ , we first observe that any  $K \in \mathcal{K}_{\overline{C}}(G)$ , does not contain any non-articulation vertex of  $K$ . This is because, as  $V(K) \cap C = \emptyset$ , any non-articulation vertex of  $K$  will remain non-dominated by the code  $C$ , a contradiction. This therefore implies that  $K$  cannot be a leaf block of  $G$  (and, in particular, the root-block of  $G$  as well). Hence, every block  $K \in \mathcal{K}_{\overline{C}}(G)$  has a positive articulation vertex  $v_K$ , say. Then we have a block  $K' \in \mathcal{K}(G)$  with its layer  $f(K') = f(K) + 1$  such that  $V(K) \cap V(K') = \{v_K\}$ . Moreover, since  $V(K) \cap C = \emptyset$ , for the code  $C$  to dominate  $v_K$ , we may assume, without loss of generality, the block  $K'$  to be such that there exists a vertex  $v_{K'}$  of  $G$  in  $V(K') \cap C$ . For every  $K \in \mathcal{K}_{\overline{C}}(G)$ , therefore, we fix such a triple  $(v_K, K', v_{K'})$  for the rest of this proof. Assume  $G[C_i]$  to be the component of  $G[C]$  such that  $v_{K'} \in C_i$ . Moreover, let  $u_i$  (the vertex of the part  $B$  of  $F_C(G)$  corresponding to  $G[C_i]$ ) belong to the component  $F_j$ , for some  $1 \leq j \leq l$ . Then, we associate  $F_j$  with the block  $K \in \mathcal{K}_{\overline{C}}(G)$ . More precisely, we define the following mapping.

$$g : \mathcal{K}_{\overline{C}}(G) \rightarrow \{F_1, F_2, \dots, F_l\}$$

$$K \mapsto F_j, \text{ where } (v_K, K', v_{K'}) \text{ is fixed, } v_{K'} \in C_i \text{ and } u_i \in F_j$$

Since the vertex  $v_{K'} \in C$  can belong to exactly one component  $G[C_i]$  of  $G[C]$  and, similarly, the vertex  $u_i$  of  $F_C(G)$  can belong to exactly one of its components  $F_j$ , the mapping  $g$  is therefore well-defined. We now claim that  $g$  is one-to-one. Indeed, consider any  $K \in \mathcal{K}_{\overline{C}}(G)$  with  $(v_K, K', v_{K'})$  associated to it, and such that  $g(K) = F_j$ . Assume by contradiction that there is  $L \in \mathcal{K}_{\overline{C}}(G)$ ,  $K \neq L$ , with  $g(L) = F_j$ , and  $(v_L, L', v_{L'})$  is associated with  $L$ . Since  $V(K) \cap C = \emptyset$ , for every vertex  $u_r \in V(F_j)$  and any block  $J \in \mathcal{I}_G(C_r)$ , we have  $f(J) \geq f(K) + 1$ . Thus,  $f(K') = f(K) + 1$  is minimum among all such blocks. By the same reasoning applied to  $L$ , we also have  $f(L') = f(L) + 1$  minimum among all blocks  $J \in \mathcal{I}_G(C_r)$ , where  $u_r \in V(F_j)$ . This implies that  $f(L') = f(K')$ , since both values are minimum among all  $f(J)$ , where  $J \in \mathcal{I}_G(C_r)$  and  $u_r \in V(F_j)$ . Therefore, we also have  $f(K) = f(L)$ . Now, since  $K \neq L$ , we must have  $V(K) \cap V(L) = \emptyset$ , or else, we would have  $|f(K) - f(L)| = 1$ , a contradiction. Now, let  $G_K$  represent the *descendant block graph* of  $K$ , that is, the connected subgraph (also a block graph) of  $G$  rooted at  $K$ . Similarly define  $G_L$  to be the descendant block graph of  $L$ . Then, by the structure of the block graph  $G$ , its subgraphs  $G_K$  and  $G_L$  are vertex-disjoint. Since  $V(K) \cap C = V(L) \cap C = \emptyset$ , all components  $G[C_r]$  of  $G[C]$  for all  $u_r \in V(F_j)$  must belong to  $G_K$  and  $G_L$  simultaneously, which is a contradiction since  $V(G_K) \cap V(G_L) = \emptyset$ . This shows that  $g$  is one-to-one, as claimed.

As noticed before, for each  $K \in \mathcal{K}_{\overline{C}}(G)$  such that  $g(K) = F_j$ , we have

$$f(K) + 1 = \min\{f(J) : J \in \mathcal{I}_G(C_r), u_r \in F_j\}.$$

Now, let  $K_0$  be the root block of  $G$ . As argued before,  $K_0 \notin \mathcal{K}_{\overline{C}}(G)$ . In other words,  $V(K_0) \cap C_{i_0} \neq \emptyset$  for some component  $G[C_{i_0}]$  of  $G[C]$ . Let  $u_{i_0} \in F_{j_0}$ . Therefore, we have

$$0 \leq \min\{f(J) : J \in \mathcal{I}_G(C_r), u_r \in F_{j_0}\} \leq f(K_0) = 0 \quad (\text{since } K_0 \in \mathcal{I}_G(C_{i_0})).$$

This implies that  $g(K) = F_{i_0}$  for any  $K$  would imply  $f(K) = -1$ , which is not possible. Hence, the image of the function  $g$  is a subset of  $\{F_1, F_2, \dots, F_l\} \setminus \{F_{i_0}\}$ . This, along with the fact that  $g$  is one-to-one, therefore implies that  $|\mathcal{K}_{\overline{C}}(G)| \leq l - 1$ .  $\square$

**Lemma 4.12.**  $|\mathcal{K}_C(G) \cup \mathcal{K}_4(G)| \leq |\mathcal{K}(G[C])| - n_0(G[C])$ .

*Proof.* Assume that  $K \in \mathcal{K}(G)$  is a block of  $\mathcal{K}_C(G) \cup \mathcal{K}_4(G)$ . Then,  $V(K)$  contains at least two vertices, say,  $u, v \in C$ . Therefore,  $uv \in E(G)$ . So, assume  $L \in \mathcal{K}(G[C])$  to be the block such that  $u, v \in V(L)$ . Then,  $V(L) \subset V(K)$ . Thus, every block  $K \in \mathcal{K}_C(G) \cup \mathcal{K}_4(G)$  can be associated with a block  $L \in \mathcal{K}(G[C])$  such that  $|V(L)| \geq 2$ . Moreover, by the structure of a block graph, this association is one-to-one. This implies that  $|\mathcal{K}_C(G) \cup \mathcal{K}_4(G)| \leq |\mathcal{K}(G[C])| - n_0(G[C])$ .  $\square$

We now prove Theorem 4.5.

**Theorem 4.5.** *Let  $G$  be a connected block graph and  $\mathcal{K}(G)$  be the set of all blocks of  $G$ . Then we have*

$$\gamma^{\text{LD}}(G) \geq \frac{|\mathcal{K}(G)| + 2}{3}.$$

*Proof.* Let  $C$  be an LD-code of the block graph  $G$ . Then, we have

$$\mathcal{K}(G) = (\mathcal{K}_C(G) \cup \mathcal{K}_4(G)) \cup \mathcal{K}_{\overline{C}}(G) \cup \mathcal{K}_2(G) \cup \mathcal{K}_3(G).$$

Therefore, using Lemma 4.9, 4.10, 4.11 and 4.12, we have

$$\begin{aligned} |\mathcal{K}(G)| &\leq |\mathcal{K}_C(G) \cup \mathcal{K}_4(G)| + |\mathcal{K}_{\overline{C}}(G)| + |\mathcal{K}_2(G)| + |\mathcal{K}_3(G)| \\ &\leq |\mathcal{K}(G[C])| - n_0(G[C]) + l - 1 + |V_2| + 2(k - l) \\ &\leq |C| - k + l - 1 + |C| + 2(k - l) && [\text{using Corollary 4.3 and Lemma 4.6}] \\ &= 2|C| + k - l - 1 \\ &\leq 3|C| - 2. \end{aligned}$$

This proves the theorem.  $\square$

Note that, for any tree  $G$  (which are particular block graphs with each block being of order 2), we have  $|\mathcal{K}(G)| = |E(G)| = |V(G)| - 1$ . The lower bounds for LD-numbers of trees given in Theorems 4.4 and 4.5 are the same; and which, in turn, are the same as that given in [185]. The following are examples of extremal block graphs described in Proposition 4.3 whose LD-numbers attain the lower bounds in Theorem 4.5: In the proof of Proposition 4.3, the graph  $G'$  obtained by deleting all the edges  $u_i u_{i+1}$  of  $G$  is an example whose LD-number ( $= k$ ) attains the bound in Theorem 4.5 (note that  $|\mathcal{K}(G')| = 3k - 2$ ). Apart from this example, there are infinite subfamilies of trees reaching the bound in Theorem 4.5 (see [185]).

### 4.1.3 Subcubic graphs

This section considers Conjecture 2.2 and its expansions for subcubic graphs. Besides proving the conjecture for subcubic graphs, we also answer the following open problems posed by Foucaud and Henning in [95]:

**Problem 4.1** ([95]). *Determine whether Conjecture 2.2 can be proven for subcubic graphs.*

**Problem 4.2** ([95]). *Determine whether Conjecture 2.2 can be proven for connected cubic graphs in general (allowing twins) with the exception of a finite set of forbidden graphs.*

As Foucaud and Henning had noted, Problem 4.2 is a weaker form of a conjecture from [127] by Henning and Löwenstein and an open problem by Henning and Rad from [128] stating that for a connected cubic graph  $G$  the LTD-number is at most half the number of vertices of  $G$ . In particular, this bound does not hold for subcubic graphs which might require two-thirds of their vertices [39]. See Figure 4.5 for an example where the  $\frac{n}{2}$ -upper bound for LTD-numbers is not true.

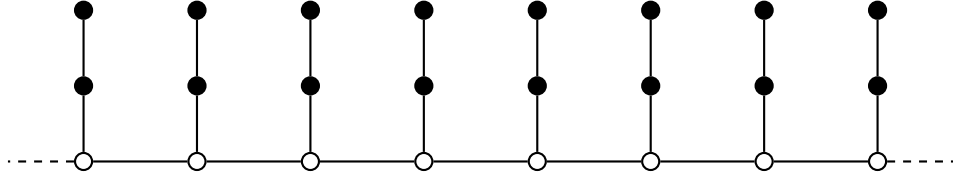


Figure 4.5: Example of a subcubic graph which shows that the  $\frac{n}{2}$ -upper bound for the LTD-number of a twin-free graph on  $n$  vertices is not true. The shaded vertices constitute a minimum LTD-code.

Coming back to our results, in Proposition 4.4, we show that a twin-free subcubic graph  $G$  has  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$ . This answers Problem 4.1 affirmatively. Then, we continue to Theorem 4.6 where we expand Proposition 4.4 by allowing subcubic graphs other than  $K_3$  and  $K_4$  to include closed twins. Finally, in Theorem 4.7, we expand the result even further to subcubic graphs containing open twins of degree 3, with the exceptions of the complete bipartite graph  $K_{3,3}$ . This answers Problem 4.2 affirmatively. Observe that our result is in fact stronger than what Problems 4.1 and 4.2 ask. We state that the  $\frac{n}{2}$ -upper bound holds for connected subcubic graphs without open twins of degrees 1 or 2 on at least four vertices with the exceptions of  $K_4$  and  $K_{3,3}$ .

We also show in Proposition 4.5 that forbidding open twins of degrees 1 and 2 is necessary. Furthermore, in Proposition 4.6, we show that Problem 4.2 cannot be expanded for  $r$ -regular graphs, for  $r \geq 4$ , with twins. We also give an infinite subfamily of twin-free subcubic graphs for which the conjecture is tight in Proposition 4.7. Note that Foucaud and Henning had asked in [95] for characterizing each twin-free cubic graph which attains the  $\frac{n}{2}$ -bound. With the help of the online graph repository [70], we have manually checked every twin-free 10-vertex cubic graph and found that there were no tight examples. Furthermore, we are aware of only one 6-vertex and one 8-vertex twin-free cubic graph (see Figures 4.6(a) and (b), also presented in [95]) whose LD-numbers attain this upper bound. From this perspective, the proof for subcubic graphs seems more challenging than the one for cubic graphs.

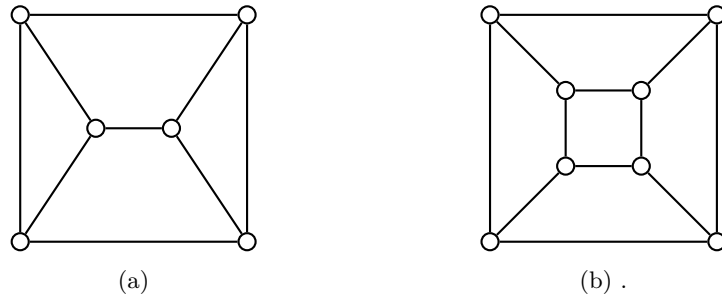


Figure 4.6: Extremal cubic graphs whose LD-numbers achieve the  $\frac{n}{2}$ -upper bound. These graphs are also presented in [95].

Results of this section have appeared in [48]. We begin with some basic observations about twin-free subcubic graphs.

**Lemma 4.13.** *Let  $G$  be a subcubic twin-free and triangle-free graph. If vertices  $u$  and  $v$  are in the same 4-cycle  $C_4$ , then all of their common neighbors are in the same cycle  $C_4$ .*

*Proof.* Assume first that  $u$  and  $v$  are adjacent. If both of them are adjacent to  $w$ , then vertices  $u, v, w$  form a triangle, a contradiction. Hence, we assume that there is a 4-cycle  $u, w, v, z$  with edges  $uw, wv, vz, zu$ . Observe that if a vertex  $b \notin \{u, w, v, z\}$  is adjacent to both  $u$  and  $v$ , then  $N_G(u) = N_G(v) = \{w, z, b\}$  since we consider a subcubic graph. This is a contradiction with  $G$  being twin-free. Hence, the claim follows.  $\square$

**Lemma 4.14.** *Let  $G$  be a subcubic twin-free graph. No two triangles in  $G$  share a common edge.*

*Proof.* Let vertices  $u, v, w$  form a triangle in  $G$ . If this triangle shares an edge with another triangle, then without loss of generality two of the vertices, say,  $v$  and  $w$  have a common neighbor  $z$ . Hence, we have  $N_G[v] = N_G[w] = \{u, v, w, z\}$ . This is a contradiction with  $G$  being a twin-free. Thus, the claim follows.  $\square$

With Lemmas 4.13 and 4.14, we obtain the following corollary.

**Corollary 4.4.** *Let  $G$  be a subcubic twin-free graph. If vertices  $v_1, v_2, v_3, v_4$  form a 4-cycle  $C_4$ , then that 4-cycle is an induced subgraph of  $G$ .*

### 4.1.3.1 Subcubic graphs with closed twins

In this section, we prove that Conjecture 2.2 holds for all subcubic graphs containing closed twins other than  $K_3$  and  $K_4$ . To get to the result, we first prove Conjecture 2.2 for isolate-free and twin-free subcubic graphs.

**Proposition 4.4.** *Let  $G$  be a twin-free and isolate-free subcubic graph on  $n$  vertices. We have*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

*Proof.* Let  $G$  be a twin-free subcubic graph on  $n$  vertices without isolated vertices. Hence,  $n \geq 4$ . If  $n = 4$ , then  $G$  is a path  $P_4$  on four vertices. We have  $\gamma^{\text{LD}}(P_4) = 2$ . Thus, the claim holds for all subcubic twin-free graphs without isolated vertices on four vertices. Suppose, to the contrary, that  $G$  is a graph with the smallest number of vertices and among those graphs one with the smallest number of edges for which the claimed upper bound does not hold. Notice that  $G$  is connected. Indeed, if there are multiple components, then the claimed upper bound does not hold on at least one of them and we could have chosen  $G$  as that component. However, this contradicts the minimality of  $G$ . We first divide the proof based on whether  $G$  has triangles.

► **Case 3:  $G$  is triangle-free.**

By Theorem 4.1, we may assume that  $G$  contains a 4-cycle. Let us call the vertices in this 4-cycle by  $a, b, c$  and  $d$  so that there are edges  $ab, bc, cd$  and  $da$ . By Corollary 4.4 there are no edges  $ac$  nor  $bd$ . Furthermore, by Lemma 4.13, the only common neighbors of vertices  $a, b, c$  and  $d$  are in the set  $\{a, b, c, d\}$ . Since  $G$  is open-twin-free, we may immediately observe that there are at most two vertices of degree 2 in a 4-cycle in  $G$ .

►► **Case 3.1: There are exactly two vertices of degree 2 in the 4-cycle.**

Observe that when there are two vertices of degree 2 in the 4-cycle, they are adjacent. Let us call these vertices, without loss of generality,  $a$  and  $b$  and denote  $G_{a,b} = G - a - b$ . Notice that  $G_{a,b}$  is connected and subcubic. We further call the other neighbor of  $b$  as  $c$  and the remaining vertex of this 4-cycle as  $d$ . Let us next divide our considerations based on whether  $G_{a,b}$  is twin-free.

►►► **Case 3.1.1:  $G_{a,b}$  is twin-free.**

Let  $S_{a,b}$  be an optimal LD-code of  $G_{a,b}$ . By the minimality of  $G$ , we have  $|S_{a,b}| \leq \frac{n}{2} - 1$ . Assume first that  $c$  or  $d$  is in  $S_{a,b}$ . We may further assume, without loss of generality that  $d \in S_{a,b}$ . Consider next  $S = S_{a,b} \cup \{a\}$  in  $G$ . We have  $a \in I_G(S, b)$  and  $b$  is the only vertex in  $V(G) \setminus S$  adjacent to



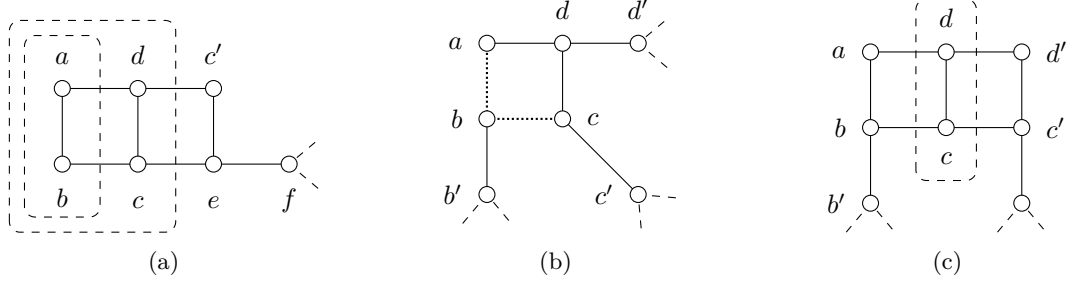


Figure 4.7: Illustrations for (a) Case 1.1.2, (b) Case 1.2.3 and (c) Case 1.2.3.2. The dotted lines indicate edges being removed. Dashed edges are edges, that may or may not exist. Dashed outlines indicate vertices being removed.

$a$ . Thus,  $I_G(S, b)$  is unique in  $V(G) \setminus S$ . Furthermore, each other vertex in  $V(G) \setminus S$  is dominated and pairwise separated from other vertices by the same vertices in  $S_{a,b}$  as in  $G_{a,b}$ . Thus, we may assume that neither of  $c$  nor  $d$  is in  $S_{a,b}$ .

Since  $S_{a,b}$  is an LD-code, vertices  $c$  and  $d$  are dominated by some other vertices in  $S_{a,b}$  which are not adjacent to  $a$  or  $b$  in  $G$ . Hence, we may again consider set  $S = S_{a,b} \cup \{a\}$  in  $G$ . Again,  $b$  is the only vertex with  $I_G(S, b) = \{a\}$  while all other vertices in  $V(G) \setminus S$  are pairwise separated and dominated by the same vertices in  $S_{a,b}$  as in  $G_{a,b}$ . Thus, when a 4-cycle contains two vertices of degree 2 in  $G$  and  $G_{a,b}$  is twin-free, we have  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$ .  $\blacktriangleleft\blacktriangleleft$

**$\blacktriangleright\blacktriangleright\blacktriangleright$  Case 3.1.2:  $G_{a,b}$  contains twins.**

Notice that since  $G$  is twin-free, at least one of the twins is  $c$  or  $d$ . First of all, if  $c$  and  $d$  are twins, then either  $G$  is a cycle on four vertices or it contains a triangle. This contradicts the twin- or triangle-freeness of  $G$ . Furthermore, if both vertices  $c$  and  $d$  are twins with  $c'$  and  $d'$ , respectively, then  $c'$  and  $d'$  have degree 2 and  $d'$  is adjacent to  $c$ , and  $c'$  is adjacent to  $d$ . Thus,  $G$  contains exactly six vertices and the set  $S = \{a, d, c'\}$  is an LD-code of  $G$  containing exactly half of the vertices in  $G$ .

Let us assume next, without loss of generality, that exactly  $c$  is a twin with vertex  $c'$ . Let us denote the other neighbor of  $c$  with  $e$  and the third neighbor of  $e$  by  $f$  (see Figure 4.7a). We have  $N_G(c') = \{d, e\}$  and  $N_G(e) = \{c, c', f\}$ . Notice that the degree of  $e$  is exactly 3 since  $G$  is subcubic and  $e$  is not a twin of  $d$ . Assume first that  $f$  is a leaf. In this case,  $G$  contains exactly seven vertices and the set  $S = \{b, d, e\}$  is an LD-code containing less than half of the vertices in  $G$ . Thus, we may assume that  $f$  is not a leaf. Consider next the graph  $G' = G_{a,b} - c - d = G - a - b - c - d$ . We notice that  $e$  is a support vertex and  $c'$  is a leaf in this graph. Moreover, since  $f$  is not a leaf and since  $G$  is twin-free, the graph  $G'$  is twin-free. By Lemmas 4.2 and 4.1, we may assume that  $S'$  is an optimal LD-code of  $G'$  such that it contains all support vertices but no leaves in  $G'$  (in particular,  $e \in S'$  and  $c' \notin S'$ ). Moreover, by the minimality of  $G$ , we have  $|S'| \leq \frac{n}{2} - 2$ . Consider next the set  $S = S' \cup \{b, d\}$ . We have  $I(a) = \{d, b\}$ ,  $I(c) = \{b, d, e\}$  and  $I(c') = \{d, e\}$ . Moreover, all other vertices in  $V(G) \setminus S$  are dominated and pairwise separated by the same vertices in  $S'$  as in  $G$ . Hence,  $S$  is locating-dominating in  $G$  with the claimed cardinality.  $\blacktriangleleft\blacktriangleleft$

Thus, if  $G$  contains two vertices of degree 2 in the same 4-cycle, then the result holds.  $\blacktriangleleft$

From now on we may assume that there is at most one vertex of degree 2 in a 4-cycle in  $G$ .

**$\blacktriangleright\blacktriangleright$  Case 3.2: There is exactly one vertex of degree 2 in the 4-cycle.**

Let us say, without loss of generality, that the vertex of degree 2 is  $a$ . Let us denote the third neighbor outside of the set  $\{a, b, c, d\}$  of  $b$  by  $b'$ , of  $c$  by  $c'$  and of  $d$  by  $d'$  (see Figure 4.7b).

**$\blacktriangleright\blacktriangleright\blacktriangleright$  Case 3.2.1: Both  $b'$  and  $d'$  are leaves.**

Let us denote  $G' = G - a - d - d'$ . Observe that  $G'$  is twin-free since  $G$  is twin-free,  $b$  is a support

vertex and  $c$  is the only non-leaf adjacent to  $b$ . Let  $S'$  be an optimal LD-code of  $G'$  which contains  $b$  but does not contain  $b'$  (such a set exists by Lemmas 4.1 and 4.2). By the minimality of  $G$ , we have  $|S'| \leq \frac{n}{2} - 1$ . Notice that to separate  $b'$  and  $c$  we have  $\{c, c'\} \cap S' \neq \emptyset$ . Hence, the set  $S = S' \cup \{d\}$  is an LD-code of  $G$ . Indeed, we have  $I(d') = \{d\}$ ,  $I(a) = \{d, b\}$ ,  $I(b') = \{b\}$  and  $|I(c)| \geq 3$  (if  $c \notin S$ ). Moreover, we have  $|S| \leq \frac{n}{2}$ .  $\blacktriangleleft\blacktriangleleft\blacktriangleleft$

**►►► Case 3.2.2: Exactly one of  $b'$  and  $d'$  is a leaf.**

Let us assume, without loss of generality, that  $b'$  is a leaf while  $d'$  is a non-leaf. In this case, we consider the graph  $G_{ab,cd} = G - ab - cd$ . Notice that this graph is twin-free since  $G$  is twin-free,  $d'$  is a non-leaf and  $c$  is the only non-leaf adjacent to  $b$  while  $b'$  is the only leaf adjacent to  $b$ . Furthermore, let  $S_{ab,cd}$  be an optimal LD-code of  $G_{ab,cd}$  such that it does not contain any leaves and contains all support vertices. It exists by Lemmas 4.1 and 4.2. Moreover, by the minimality of  $G$  we have  $|S_{ab,cd}| \leq \frac{n}{2}$ . In particular, we have  $b, d \in S_{ab,cd}$ . Furthermore, since  $I_{G_{ab,cd}}(S_{ab,cd}, c) \neq I_{G_{ab,cd}}(S_{ab,cd}, b')$ , we have  $c \in S_{ab,cd}$  or  $c' \in S_{ab,cd}$ . Let us next consider set  $S_{ab,cd}$  in  $G$ . Notice that  $I_G(S_{ab,cd}, a) = \{b, d\}$ ,  $I_G(S_{ab,cd}, b') = \{b\}$ ,  $|I_G(S_{ab,cd}, c)| \geq 3$  (if  $c \notin S_{ab,cd}$ ) and  $b$  separates the vertices  $d'$  and  $c$ . Thus,  $S_{ab,cd}$  is an LD-code of claimed cardinality in  $G$ .  $\blacktriangleleft\blacktriangleleft\blacktriangleleft$

From now on we assume that when a 4-cycle has exactly one degree 2 vertex, neither neighbor of the degree two vertex is a support vertex.

**►►► Case 3.2.3: Neither  $b'$  nor  $d'$  is a leaf.**

Let us next consider graph  $G_{ab,bc} = G - ab - bc$  (see Figure 4.7b). We further divide the proof based on three possibilities: Either  $G_{ab,bc}$  is twin-free, or vertex  $b$  is a twin with some other vertex or vertex  $c$  is a twin with some other vertex. There cannot exist any other twins in  $G_{ab,bc}$ . Indeed,  $G$  is twin-free,  $d$  is the support vertex adjacent to  $a$  and the leaf  $a$  is not a twin since  $d'$  is not a leaf.

**►►►► Case 3.2.3.1:  $G_{ab,bc}$  is twin-free.**

Let  $S_{ab,bc}$  be an optimal LD-code which does not contain any leaves and contains all the support vertices in  $G_{ab,bc}$ . The set  $S_{ab,bc}$  exists by Lemmas 4.2 and 4.1. Furthermore, by the minimality of  $G$  it has cardinality of at most  $\frac{n}{2}$ . Observe that  $d \in S_{ab,bc}$  and  $b' \in S_{ab,bc}$ . Observe further that if  $c \in S_{ab,bc}$  and  $c' \notin S_{ab,bc}$ , then  $S'_{ab,bc} = (S_{ab,bc} \setminus \{c\}) \cup \{c'\}$  is also an LD-code of  $G_{ab,bc}$ . Indeed,  $c$  is the only vertex with  $I(S'_{ab,bc}, c) = \{d, c'\}$  since  $I(a) = \{d\}$  and  $|I(S_{ab,bc}, d')| \geq 2$  so if  $c' \in N_G(d')$ , then  $|I(S'_{ab,bc}, d')| \geq 3$  and  $d'$  is separated from all other vertices.

We claim that  $S_{ab,bc}$  or  $S'_{ab,bc}$  is an LD-code of claimed cardinality in  $G$ . Let us first consider the set  $S_{ab,bc}$ . Since  $I_G(S_{ab,bc}, a) = \{d\}$ , the only  $I$ -set which might change when we consider  $G$  instead of  $G_{ab,bc}$  is  $I(b)$ . We have  $I_G(S_{ab,bc}, b) = \{b'\}$  or  $I_G(S_{ab,bc}, b) = \{b', c\}$ . If  $I_G(S_{ab,bc}, b) = \{b'\}$ , then no  $I$ -set is modified when we change the perspective from  $G_{ab,bc}$  to  $G$  and in this case set  $S_{ab,bc}$  is an LD-code of  $G$ . On the other hand, if  $I_G(S_{ab,bc}, b) = \{b', c\}$ , then it is possible that  $I_G(S_{ab,bc}, b) = I_G(S_{ab,bc}, c')$ . If this is not the case, then  $S_{ab,bc}$  is an LD-code of  $G$ . However, if  $I(b) = I(c')$ , then  $c' \notin S_{ab,bc}$  and we may consider the set  $S'_{ab,bc}$ . Notice that this change does not modify  $I(a)$ . Moreover, we have  $I_G(S'_{ab,bc}, b) = \{b'\}$ . Since  $S'_{ab,bc}$  is an LD-code of  $G_{ab,bc}$  and no  $I$ -sets are modified when we transfer to  $G$ , the set  $S'_{ab,bc}$  is an LD-code also of  $G$ . Moreover, both  $S_{ab,bc}$  and  $S'_{ab,bc}$  satisfy the claimed upper bound on the cardinality.  $\blacktriangleleft\blacktriangleleft\blacktriangleleft\blacktriangleleft$

**►►►► Case 3.2.3.2:  $c$  is a twin in  $G_{ab,bc}$ .**

Notice that since  $c$  is adjacent to exactly  $d$  and  $c'$  in  $G_{ab,bc}$  and  $N_G(d) = \{a, c, d'\}$ , the vertex  $c$  is a twin with  $d'$ . Notice that since  $c, c', d', d$  is a 4-cycle and  $d'$  has degree 2 in  $G$ , we have  $\deg(c') = 3$  and  $c'$  is not a support vertex (see Figure 4.7c). Let us next consider the graph  $G_{c,d} = G - c - d$ . Since  $G$  is twin-free and neither  $c'$  nor  $b$  is a support vertex in  $G$ , the graph  $G_{c,d}$  is twin-free. Hence, there exists an optimal LD-code  $S_{c,d}$  containing no leaves and every support vertex in  $G_{c,d}$  by Lemmas 4.2 and 4.1 with cardinality  $|S_{c,d}| \leq \frac{n}{2} - 1$ . Notice that  $b, c' \in S_{c,d}$ . Let us consider the set  $S = S_{c,d} \cup \{d\}$ . We have  $I_G(S, a) = \{b, d\}$ ,  $I_G(S, c) = \{b, d, c'\}$  and  $I_G(S, d') = \{c', d\}$ . All other  $I$ -sets remain unmodified and since  $S_{c,d}$  is an LD-code of  $G_{c,d}$ , the set  $S$  is locating-dominating in  $G$ .  $\blacktriangleleft\blacktriangleleft\blacktriangleleft\blacktriangleleft$

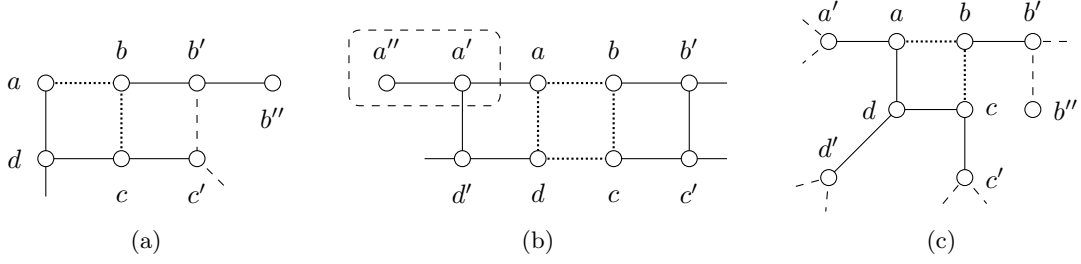


Figure 4.8: Illustrations for (a) Case 1.2.3.3, (b) Case 1.3.2 and (c) Case 1.3.3. The dotted edges indicate edges being removed. Dashed edges are edges, that may or may not exist. Dashed outlines indicate vertices being removed.

►►►► **Case 3.2.3.3:  $b$  is a twin in  $G_{ab,bc}$ .**

Notice that since  $b$  is a leaf in  $G_{ab,bc}$ , there is a leaf adjacent to  $b'$  in  $G$ . Let us call this leaf  $b''$  (see Figure 4.8a).

Assume first that there is an edge from  $b'$  to  $c'$ . In this case, we consider the graph  $G_{b',b''} = G - b' - b''$ . This graph is twin-free since  $\deg(d) = \deg(c) = 3$  and hence, neither  $b$  nor  $c'$  may become a twin with a removal of their neighbor. Hence, there exists an optimal LD-code  $S_{b',b''}$  containing no leaves and every support vertex in  $G_{b',b''}$  by Lemmas 4.2 and 4.1 with cardinality  $|S_{b',b''}| \leq \frac{n}{2} - 1$ . Furthermore, the set  $S = S_{b',b''} \cup \{b''\}$  is an LD-code of cardinality at most  $\frac{n}{2}$  in  $G$ . Indeed, each  $I$ -set of a vertex in  $V(G_{b',b''})$  remains unmodified while  $b'' \in I(S, b')$  and no other vertex is adjacent to  $b''$ . Thus, we may assume that edge  $b'c'$  does not exist.

We may now consider the graph  $G_{bb'} = G - bb'$ . Observe that either  $G_{bb'}$  is twin-free or vertices  $b'$  and  $b''$  form a two-vertex component  $P_2$ . Since  $\gamma^{\text{LD}}(P_2) = 1$ , there exists an LD-code  $S_{bb'}$  in  $G_{bb'}$  which has cardinality at most  $\frac{n}{2}$  and contains all support vertices and no leaves in  $G_{bb'}$  by Lemmas 4.2 and 4.1 (if  $b'$  and  $b''$  form a  $P_2$  component we consider  $b'$  as a support vertex and  $b''$  as a leaf). In particular, we have  $b' \in S_{bb'}$ . Hence, when we consider  $S_{bb'}$  in  $G$ , the only  $I$ -set which changes between  $G$  and  $G_{bb'}$  is  $I(b)$ . Hence, we only need to confirm that  $I(b)$  is unique when  $b \notin S_{bb'}$ . First of all, observe that  $|I_G(b)| \geq 2$ . If  $a \in I_G(b)$ , then  $I_G(b)$  is unique since the edge  $db'$  does not exist by Lemma 4.13. Thus, we may assume that  $c \in I_G(b)$ . Hence, if  $I_G(b) = I_G(x)$ , then  $x \in N_G(c)$  and  $x = d$  or  $x = c'$ . However, by Lemma 4.13, we have  $d \notin N_G(b')$ . Thus,  $x = c'$ . However, by our assumption, the edge  $c'b'$  does not exist. Therefore,  $I_G(b)$  is unique and  $S_{bb'}$  is an LD-code of  $G$  with the claimed cardinality. ◀◀◀◀

Hence, the result holds when neither  $b'$  nor  $d'$  is a leaf ◀◀◀

Thus, the result holds when there is exactly one vertex of degree 2 in the 4-cycle. ◀◀

Therefore, we may assume from now on that the 4-cycle contains no vertices of degree 2.

►► **Case 3.3: Every vertex in a 4-cycle has degree 3.**

We denote the neighbors of  $a, b, c$  and  $d$  that are outside of the 4-cycle by  $a', b', c'$  and  $d'$ , respectively. Recall that due to Lemma 4.13 the vertices  $a', b', c'$  and  $d'$  are distinct. We divide the proof into cases based on which of the possible edges between  $a', b', c'$  and  $d'$  are present.

►►► **Case 3.3.1: The edges  $a'b', b'c', c'd'$  and  $d'a'$  are present in  $G$ .**

The entire graph  $G$  is now determined. Indeed, we have  $G = P_2 \square C_4$  and the conjectured bound holds, since  $G$  is a cubic graph (see [95]). ◀◀◀

Therefore, we may assume that at least one of the edges in the subcase above is not present. Without loss of generality, we assume that the edge  $a'b'$  is not present. The proof is then divided into cases based on whether the incident edges  $b'c'$  and  $a'd'$  are present in  $G$ .

►►► **Case 3.3.2: The edge  $a'b'$  is not present but the edges  $b'c'$  and  $a'd'$  are present in  $G$ .**

If none of the vertices  $a'$ ,  $b'$ ,  $c'$  and  $d'$  are support vertices, then the graph  $G' = G - ab - bc - cd - da$  is clearly twin-free (see Figure 4.8b). The vertices  $a$ ,  $b$ ,  $c$  and  $d$  are leaves, and the vertices  $a'$ ,  $b'$ ,  $c'$  and  $d'$  are support vertices in  $G'$ . By the minimality of the number of edges of  $G$  (or by the fact that  $G'$  is now  $C_4$ -free), there exists an LD-code  $S'$  of  $G'$  such that  $|S'| \leq \frac{n}{2}$ . Due to Lemmas 4.1 and 4.2, we may assume that  $a', b', c', d' \in S'$  and  $a, b, c, d \notin S'$ . The  $I$ -sets given by  $S'$  are identical in  $G'$  and  $G$ , and thus  $S'$  is an LD-code of  $G$  with  $|S'| \leq \frac{n}{2}$ .

Assume then that at least one of  $a'$ ,  $b'$ ,  $c'$  and  $d'$ , say  $a'$ , is a support vertex. (Notice that if  $c'$  or  $d'$  is a support vertex, then the edge  $c'd'$  is not present, and these cases are symmetrical to  $a'$  being a support vertex.) Let  $a''$  be the leaf attached to  $a'$  (see Figure 4.8b). Consider the graph  $G_{a',a''} = G - a' - a''$ . The only vertices that might have twins in  $G_{a',a''}$  are  $a$  and  $d'$ . The vertex  $a$  does not have a twin, since  $N_{G_{a',a''}}(a) = \{b, d\}$  and the only other vertex adjacent to both  $b$  and  $d$  is  $c$ , but  $c'$  is a neighbor of  $c$  that is not adjacent to  $a$ . If  $d'$  has a twin, then that twin must be  $a$ ,  $c$  or  $d$ . The vertex  $a$  has no twins,  $b$  is adjacent to  $c$  but not  $d'$ , and  $c$  is adjacent to  $d$  but not  $d'$ . Thus, the vertex  $d'$  has no twins either. Therefore, the graph  $G_{a',a''}$  is twin-free. By the minimality of  $G$ , there exists an LD-code  $S_{a',a''}$  of  $G_{a',a''}$  with cardinality at most  $\frac{n}{2} - 1$ . Now the set  $S = S_{a',a''} \cup \{a'\}$  is an LD-code of  $G$  since  $a''$  is the only vertex with an  $I$ -set containing only  $a'$ . Since  $|S| \leq \frac{n}{2}$ , the claim holds. ◀◀◀

►►► **Case 3.3.3: The edges  $a'b'$  and  $b'c'$  are not present in  $G$ .**

Consider the graph  $G_{ab,bc} = G - ab - bc$ . The vertex  $b$  is a leaf, and  $b'$  is a support vertex in  $G_{ab,bc}$  (see Figure 4.8c).

►►►► **Case 3.3.3.1:  $G_{ab,bc}$  is twin-free.**

There exists an LD-code  $S_{ab,bc}$  of  $G_{ab,bc}$  such that  $b' \in S_{ab,bc}$  and  $b \notin S_{ab,bc}$  (by Lemmas 4.1 and 4.2), and  $|S_{ab,bc}| \leq \frac{n}{2}$ . Since  $b \notin S_{ab,bc}$ ,  $I(b)$  is the only  $I$ -set given by  $S_{ab,bc}$  that can be different in  $G$  when compared to  $G_{ab,bc}$ . Indeed, if  $a, c \notin S_{ab,bc}$ , then all  $I$ -sets in  $G$  are identical to the  $I$ -sets in  $G_{ab,bc}$ . If  $a \in S_{ab,bc}$  or  $c \in S_{ab,bc}$ , then  $I_G(b)$  contains  $a$  or  $c$ , but the rest of the  $I$ -sets are identical to those of  $G_{ab,bc}$ . The only vertices whose  $I$ -sets could be the same as  $I_G(b)$  are  $a'$ ,  $c'$  and  $d$ , but  $b' \in I_G(b)$  and  $b'$  is not adjacent to  $a'$ ,  $c'$  or  $d$  (due to the edges  $a'b'$  and  $b'c'$  not being present and Lemma 4.13). Thus,  $I_G(b)$  is unique and  $S_{ab,bc}$  is an LD-code of  $G$  with  $|S_{ab,bc}| \leq \frac{n}{2}$ . ◀◀◀◀

►►►► **Case 3.3.3.2:  $G_{ab,bc}$  is not twin-free.**

Now at least one of  $a$ ,  $b$  and  $c$  has a twin. Suppose that  $a$  has a twin. Since  $N_{G_{ab,bc}}(a) = \{a', d\}$ ,  $d'$  and  $c$  are the only possible twins of  $a$ . If  $d'$  is a twin with  $a$ , then  $aa'd'da$  is a cycle in  $G$  and the degree of  $d'$  is two in  $G$ . This contradicts our assumption that all vertices in a 4-cycle have degree 3 in  $G$ . If  $c$  is twins with  $a$ , then  $a' = c'$ , but this is impossible due to Lemma 4.13. Thus, neither  $a$  nor  $c$  (by symmetry) have twins in  $G_{ab,bc}$ . Therefore,  $b$  has a twin in  $G_{ab,bc}$ . Now either  $b$  and  $b'$  form a  $P_2$  component in  $G_{ab,bc}$  or  $b'$  has a leaf  $b''$  in  $G$ . These cases are handled somewhat similarly as Case 3.2.3.3.

Assume that  $b$  and  $b'$  form a  $P_2$ -component in  $G_{ab,bc}$ . The graph  $G_{b,b'} = G - b - b'$  is twin-free, and thus there exists an LD-code  $S_{b,b'}$  such that  $|S_{b,b'}| \leq \frac{n}{2} - 1$ . The set  $S = S_{b,b'} \cup \{b\}$  is clearly an LD-code of  $G$ , and we have  $|S| \leq \frac{n}{2}$ .

Assume then that  $b'$  has a leaf  $b''$  in  $G$ . Consider the graph  $G_{bb'} = G - bb'$ . Again,  $G_{bb'}$  is either twin-free or  $b'$  and  $b''$  form a  $P_2$ -component in  $G_{bb'}$ . As in the previous case, if  $b'$  and  $b''$  form a  $P_2$ -component in  $G_{bb'}$ , we can easily construct an LD-code  $S$  of  $G$  such that  $|S| \leq \frac{n}{2}$  by considering an LD-code of  $G - b' - b''$ . So assume that  $G_{bb'}$  is twin-free. There exists an LD-code  $S_{bb'}$  of  $G_{bb'}$  such that  $|S_{bb'}| \leq \frac{n}{2}$  and  $b' \in S_{bb'}$ . We claim that the set  $S_{bb'}$  is also an LD-code of  $G$ . The only  $I$ -set that might differ between the two graphs is  $I(b)$  (assuming  $b \notin S_{bb'}$ ). Since  $S_{bb'}$  is an LD-code of  $G_{bb'}$ , we have  $a \in S_{bb'}$  or  $c \in S_{bb'}$ . Now, if  $I_G(b)$  is the same as the  $I$ -set of some other vertex, then that vertex must be  $a'$ ,  $d$  or  $c'$ . However, we also have  $b' \in I_G(b)$  and  $b'$  is not adjacent to  $a'$ ,  $d$  or  $c'$ . Thus,  $I_G(b)$  is unique, and the set  $S_{bb'}$  is an LD-code of  $G$  with  $|S_{bb'}| \leq \frac{n}{2}$ . ◀◀◀◀

Thus, we have the result when the edges  $a'b'$ ,  $b'c'$ ,  $c'd'$  and  $d'a'$  are present in  $G$ . ◀◀◀

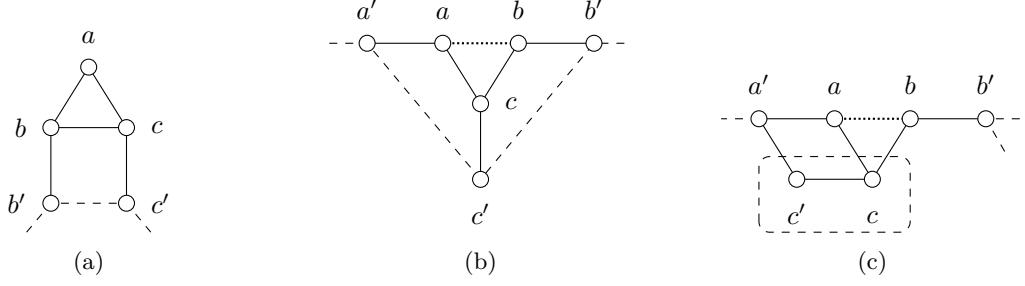


Figure 4.9: Illustrations for (a) Case 2.1, (b) Case 2.2 and (c) Case 2.2.2. The dotted edges indicate edges being removed. Dashed edges are edges, that may or may not exist. Dashed outlines indicate vertices being removed.

This implies that the result is true when every vertex in the 4-cycle is of degree 3.  $\blacktriangleleft\blacktriangleleft$

Therefore,  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$  for all triangle-free twin-free subcubic graphs with no isolated vertices.  $\blacktriangleleft$

We then assume that  $G$  is not triangle-free.

► **Case 4:  $G$  has triangles as induced subgraphs.**

Let us assume  $T = G[a, b, c]$  to be a triangle in  $G$  induced by the vertices  $a$ ,  $b$  and  $c$ . If any two vertices of  $T$  are of degree 2 in  $G$ , then it implies that the said vertices are twins in  $G$ , a contradiction to our assumptions. Hence, we assume from here on that at most one vertex of  $T$  is of degree 2 in  $G$ .

►► **Case 4.1:  $T$  has a vertex of degree 2 in  $G$ .**

Without loss of generality, let us assume that  $\deg_G(a) = 2$ . This implies that the vertices  $b$  and  $c$  have neighbors, say  $b'$  and  $c'$ , respectively, in  $G$  outside of  $T$  (see Figure 4.9a). Observe that  $b' \neq c'$ , or else, the pair  $b$  and  $c$  would be twins in  $G$ , a contradiction to our assumption. Let  $G_{ab} = G - ab$ .

►►► **Case 4.1.1:  $G_{ab}$  is twin-free.**

By our assumption on the minimality of the graph  $G$ , there exists an LD-code  $S_{ab}$  of  $G_{ab}$  such that  $|S_{ab}| \leq \frac{n}{2}$ . Moreover, by Lemmas 4.1 and 4.2, since  $c$  is a support vertex and  $a$  is a leaf in  $G_{ab}$ , we can assume that  $c \in S_{ab}$  and  $a \notin S_{ab}$ . We then show that the set  $S_{ab}$  is also an LD-code of  $G$ . If  $b \notin S_{ab}$ , then  $I_G(x) = I_{G_{ab}}(x)$  for each  $x \in V(G) \setminus S_{ab}$ , and thus  $S_{ab}$  is an LD-code of  $G$ .

Let us, therefore, assume next that  $b \in S_{ab}$ . Now, if  $S_{ab}$  is not an LD-code of  $G$ , it would imply that there exists a vertex  $x$  of  $G$  other than  $a$  and not in  $S_{ab}$  such that the pair  $a, x$  is separated in  $G_{ab}$  but not in  $G$ . Since  $I_G(a) = \{b, c\}$ , we must have  $I_G(x) = \{b, c\}$  which makes the vertices  $b$  and  $c$  twins in  $G$ , a contradiction to our assumption. Hence,  $S_{ab}$  is an LD-code of  $G$  also in the case that  $a \notin S_{ab}$  and  $b \in S_{ab}$ .

Thus, overall,  $S_{ab}$  is an LD-code of  $G$  with  $|S_{ab}| \leq \frac{n}{2}$  and thus, the result follows in the case that the graph  $G_{ab}$  is twin-free.  $\blacktriangleleft\blacktriangleleft$

Now, by symmetry, we may assume that, if the graph  $G_{ac} = G - ac$  is also twin-free, then the result holds as well.

►►► **Case 4.1.2: Both  $G_{ab}$  and  $G_{ac}$  have twins.**

Let us first look at the graph  $G_{ab}$ . The twins in  $G_{ab}$  must either be a pair  $a, x$  or a pair  $b, y$ , where  $x$  and  $y$  are vertices of  $G$  different from  $a$  and  $b$ , respectively.

►►►► **Case 4.1.2.1:  $a, x$  are twins in  $G_{ab}$  for some  $x \in V(G) \setminus \{a\}$ .**

In this case, since  $ac \in E(G)$ , we must have  $x \in N_G(c) \setminus \{a\} = \{b, c'\}$ . If  $x = b$ , that is, if  $a$  and  $b$  are twins in  $G_{ab}$ , it implies a contradiction since  $\deg_{G_{ab}}(a) = 1 \neq 2 = \deg_{G_{ab}}(b)$ . Therefore,  $x \neq b$ . In other words, we have  $x = c'$ , that is,  $a$  and  $c'$  are twins in  $G_{ab}$ . Therefore, we must also have

$\deg_G(c') = \deg_{G_{ab}}(c') = \deg_{G_{ab}}(a) = 1$ . We now look at the graph  $G_{ac} = G - ac$ . By our assumption on Case 4.1.2, the graph  $G_{ac}$  also has twins. However,  $G_{ac}$  cannot have twins of the form  $c, z$  for some vertex  $z \neq c$  of  $G$ , since the neighbor  $c'$  of  $c$  is of degree 1 in  $G_{ac}$ . Therefore, by analogy to the previous case when  $a$  and  $c'$  were twins in  $G_{ab}$ , the vertices  $a$  and  $b'$  must be twins in  $G_{ac}$ . Again, by the same analogy, we infer that  $\deg_G(b') = 1$ . Therefore, the graph  $G$  is determined and it can be checked that the set  $S = \{b, c\}$  is an LD-code of  $G$  with  $|S| < \frac{1}{2} \times 5 = \frac{n}{2}$ . Hence, the result follows.  $\blacktriangleleft\blacktriangleleft\blacktriangleleft$

Again, by symmetry, we may assume that the result also holds in the case where  $a$  is a twin in graph  $G_{ac}$ . Thus, we assume from here on that in graphs  $G_{ab}$  or  $G_{ac}$  the vertex  $a$  does not belong to a pair of twins.

**►►► Case 4.1.2.2:  $b, y$  are twins in  $G_{ab}$  for some vertex  $y \in V(G) \setminus \{a, b\}$ .**

In this case, since  $bc \in E(G)$ , we must have  $y \in N_G(c) \setminus \{a, b\} = \{c'\}$ . Therefore, we must have  $y = c'$  with  $b'c' \in E(G)$ . Moreover,  $\deg_G(c') = \deg_{G_{ab}}(c') = \deg_{G_{ab}}(b) = 2$ . We again look at the graph  $G_{ac}$  which, by assumption on Case 4.1.2, has twins other than the pair  $a, b'$ . Therefore, by symmetry to the graph  $G_{ab}$ ,  $b'$  and  $c$  must be twins in  $G_{ac}$  with  $\deg_G(b') = 2$ . Therefore, the graph  $G$  is again determined and it can be checked that the set  $S = \{b, c\}$  is an LD-code of  $G$  with  $|S| = 2 < \frac{1}{2} \times 5 = \frac{n}{2}$ . Hence, the result holds in this case.  $\blacktriangleleft\blacktriangleleft\blacktriangleleft$

Hence, the result follows in the case that both  $G_{ab}$  and  $G_{ac}$  have twins.  $\blacktriangleleft\blacktriangleleft$

In conclusion, therefore, the claim holds when  $T$  has a vertex of degree 2 in  $G$ .  $\blacktriangleleft\blacktriangleleft$

**►► Case 4.2: Each vertex of  $T$  is of degree 3 in  $G$ .**

By assumption, we have  $\deg_G(a) = \deg_G(b) = \deg_G(c) = 3$ . Let  $N_G(a) \setminus \{b, c\} = a'$ ,  $N_G(b) \setminus \{a, c\} = b'$  and  $N_G(c) \setminus \{a, b\} = c'$ . Notice that each of  $a'$ ,  $b'$  and  $c'$  must be distinct, or else,  $G$  would have twins, a contradiction to our assumption. If  $a'b', a'c', b'c' \in E(G)$ , then the graph is a cubic graph in which the vertex subset  $S = \{a, b, c\}$  can be checked to be an LD-code of order  $|S| = 3 = \frac{1}{2} \times 6 = \frac{n}{2}$ . Hence, in this case, the result holds. Let us, therefore, assume without loss of generality that  $a'b' \notin E(G)$  (see Figure 4.9b). We now consider the graph  $G_{ab} = G - ab$ .

**►►► Case 4.2.1:  $G_{ab}$  is twin-free.**

By the minimality of  $G$ , let us assume that  $S_{ab}$  is an LD-code of  $G_{ab}$  such that  $|S_{ab}| \leq \frac{n}{2}$ . We show that the set  $S_{ab}$  is also an LD-code of  $G$ . Now, if either  $a, b \in S_{ab}$  or  $a, b \notin S_{ab}$ , then the set  $S_{ab}$  is also an LD-code of  $G$  and we are done. Indeed, in these cases we have  $I_G(S_{ab}; x) = I_{G_{ab}}(S_{ab}; x)$  for each  $x \notin S_{ab}$ . Hence, by symmetry and therefore without loss of generality, let us assume that  $a \in S_{ab}$  and  $b \notin S_{ab}$ . Let us next suppose, to the contrary, that  $S_{ab}$  is not an LD-code of  $G$ . Then, the only way that can happen is if there exists a vertex  $y$  of  $G$  other than  $b$  and not in  $S_{ab}$  such that the pair  $b, y$  is separated in  $G_{ab}$  but not in  $G$ . In particular, we must have  $y \in N_{G_{ab}}(a) = \{a', c\}$ . Let us first assume that  $y = a'$ , that is, the pair  $a', b$  is not separated by  $S_{ab}$  in  $G$ . We must have  $|\{b', c\} \cap S_{ab}| \geq 1$  in order for  $S_{ab}$  to dominate  $b$ . Now, if  $b' \in S_{ab}$ , it implies that  $a'b' \in E(G)$  contrary to our assumption. Therefore,  $b' \notin S_{ab}$ . This implies that  $c \in S_{ab}$ . This further implies that  $a'c \in E(G)$ , thus making the pair  $a, c$  twins in  $G$ , a contradiction to our assumption. Hence, we conclude that  $y \neq a'$ . Let us therefore assume now that  $y = c$ , that is the pair  $b, c$  is not separated by  $S_{ab}$  in  $G$ . Therefore, in particular, we have  $c \notin S_{ab}$ . Therefore, in order for  $S_{ab}$  to dominate  $b$ , we must have  $b' \in S_{ab}$  which implies that  $b'c \in E(G)$ . This further implies that the vertices  $b$  and  $c$  are twins in  $G$ , a contradiction to our assumption. Therefore,  $y \neq c$  either. In other words, this proves that  $S_{ab}$  is indeed an LD-code of  $G$ .

Hence, in the case that the graph  $G_{ab}$  is twin-free, we find an LD-code  $S_{ab}$  of  $G$  such that  $|S_{ab}| \leq \frac{n}{2}$  and thus, the result holds.  $\blacktriangleleft\blacktriangleleft\blacktriangleleft$

**►►► Case 4.2.2:  $G_{ab}$  has twins.**

In this case, a twin pair in  $G_{ab}$  is either of the form  $a, x$  or of the form  $b, y$ , where  $x$  and  $y$  are vertices of  $G$  different from  $a$  and  $b$ , respectively. By symmetry, let us consider a pair  $a, x$  to be twins in  $G_{ab}$ . Now, since  $ac \in E(G)$ , we must have  $x \in N_G(c) \setminus \{a\} = \{b, c'\}$ . If  $x = b$ , that is, if  $a$  and  $b$

are twins in  $G_{ab}$ , then they are twins in  $G$  as well, a contradiction to our assumption. Therefore,  $x \neq b$  and so, we have  $x = c'$ , that is,  $a$  and  $c'$  are twins in  $G_{ab}$ . This implies that  $a'c' \in E(G)$  (see Figure 4.9c). Moreover,  $\deg_G(c') = \deg_{G_{ab}}(c') = \deg_{G_{ab}}(a) = 2$ . We next consider the graph  $G_{c,c'} = G - c - c'$ .

Observe first that  $G_{c,c'}$  is connected since  $N_G(c') = \{a', c\}$  and  $N_G(c) = \{a, b, c'\}$ . Furthermore, graph  $G_{c,c'}$  is also twin-free. Indeed, if  $x \in V(G_{c,c'})$  is a twin with some vertex  $y \in V(G_{c,c'})$ , then we may assume, without loss of generality, that  $x \in \{a, b, a'\}$ . We have  $N_{G_{c,c'}}(a) = \{b, a'\}$ . Thus, if  $x = a$ , then  $y = b'$  and the edge  $a'b' \in E(G)$ , a contradiction. By symmetry, we have that  $x \neq b$  and  $y \notin \{a, b\}$  and hence,  $x = a'$ . However, now  $y \in N_{G_{c,c'}}(a)$ . Thus,  $y = b$ , a contradiction. Therefore, graph  $G_{c,c'}$  is twin-free and, by the minimality of  $G$ , it admits an LD-code  $S_{c,c'}$  of cardinality at most  $\frac{n}{2} - 1$ .

Observe that for set  $S_{c,c'}$  to dominate  $a$  in  $G_{c,c'}$ , we have  $\{a, b, a'\} \cap S_{c,c'} \neq \emptyset$ . First assume that  $a' \in S_{c,c'}$ .

►►►► **Case 4.2.2.1:**  $a' \in S_{c,c'}$ .

We consider set  $S = S_{c,c'} \cup \{c'\}$ . In particular, set  $S$  is dominating in  $G$  and it separates all vertex pairs  $x, y \in V(G_{c,c'}) \setminus S_{c,c'}$  using the vertices in  $S_{c,c'}$ . Furthermore, vertex  $c$  is the only vertex in  $V(G) \setminus S_{c,c'}$  adjacent to  $c' \in S$ . Thus,  $S$  separates all vertices in  $V(G) \setminus S$  and it is an LD-code in graph  $G$  with cardinality at most  $\frac{n}{2}$ . ◀◀◀◀

Assume next that  $a$  or  $b$  is in  $S_{c,c'}$ .

►►►► **Case 4.2.2.2:**  $a \in S_{c,c'}$  or  $b \in S_{c,c'}$ .

We consider set  $S = S_{c,c'} \cup \{c\}$ . Clearly,  $S$  is dominating in  $G$ . Furthermore, as in the previous case, set  $S$  separates all vertices  $x, y \in V(G_{c,c'}) \setminus S_{c,c'}$  using the vertices in  $S_{c,c'}$ . Finally, vertex  $c'$  is adjacent to  $c$  but neither of the vertices  $a$  nor  $b$ . Thus, also  $c'$  is separated from all other vertices in  $V(G) \setminus S$ . Hence,  $S$  separates all vertices in  $V(G) \setminus S$  and it is an LD-code in graph  $G$  with cardinality at most  $\frac{n}{2}$ . ◀◀◀◀

This implies that the graph  $G$  admits an LD-code of the claimed cardinality in the case that  $G_{ab}$  has twins. ◀◀◀

Thus, in the case that all three vertices of  $T$  are of degree 3 in  $G$ , the result follows. ◀◀

Therefore, the theorem holds if the graph  $G$  has triangles as induced subgraphs. ◀

Finally, Cases 1 and 2 together prove the theorem. ◻

We next prove the main theorem of this section.

**Theorem 4.6.** *Let  $G$  be a connected open-twin-free and isolate-free subcubic graph on  $n$  vertices other than  $K_3$  and  $K_4$ . Then, we have*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

*Proof.* Let us assume, to the contrary, that there exists an open-twin-free subcubic graph  $G$  on  $n$  vertices without isolated vertices other than  $K_3$  or  $K_4$  for which we have  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Furthermore, let us assume that  $G$  has the fewest number of closed twins among these graphs. By Proposition 4.4, there is at least one pair of closed twins in  $G$ . Notice that since  $G$  is not  $K_4$ , we cannot have a triple of pairwise closed twins. Moreover, notice that closed twins have degree 2 or 3.

Let us first assume that there exist closed twins  $u, v$  of degree 2 in  $G$  and that they are adjacent to vertex  $w$ . Notice that now  $\deg(w) = 3$ . Consider graph  $G' = G - vw$ . In this graph, vertex  $v$  is a leaf, there are no open twins and the number of closed twins is smaller than in  $G$ . Hence,  $\gamma^{\text{LD}}(G') \leq \frac{n}{2}$ . Let us denote by  $S'$  an optimal LD-code of  $G'$ . By Lemma 4.1, we may assume that  $u \in S'$  and by Lemma 4.2, we may assume that  $v \notin S'$ . Moreover, since  $I_{G'}(w) \neq I_{G'}(v) = \{u\}$ , we have some vertex  $z \in I_{G'}(w)$ . However, now  $S'$  is an LD-code of  $G$ . Indeed, we either have  $z \neq w$

and  $I_G(v) = \{u\}$  while  $I_G(w) = \{z, u\}$ . If on the other hand  $z = w$ , then  $v$  is separated from all other vertices in  $V(G) \setminus S'$  since  $v$  is the only vertex in  $V(G) \setminus S'$  which is adjacent to  $u$ .

Let us next assume that  $u$  and  $v$  are closed twins of degree 3 in  $G$ . Let us denote  $N_G[u] = N_G[v] = \{u, v, w, w'\}$ . Notice that if there is an edge between  $w$  and  $w'$ , then  $G$  is the complete graph  $K_4$ . Moreover, if they are adjacent to the same vertex  $z \neq u, v$ , then  $w$  and  $w'$  are open twins. The same is true if  $w$  and  $w'$  have degree 2. Hence, we may assume that  $z \in N_G(w) \setminus N_G(w')$ . Assume next that  $\deg(w') = 2$ . First notice that if  $z$  is a leaf, then  $G$  has five vertices and set  $\{w, v\}$  is locating-dominating in  $G$ . Hence, we may assume that  $z$  is not a leaf. Consider graph  $G_{v,w'} = G - v - w'$ . Since  $G_{v,w'}$  has fewer closed twins than  $G$ , there exists an LD-code  $S_{v,w'}$  of cardinality at most  $\frac{n}{2} - 1$  which has  $w \in S_{v,w'}$  and  $u \notin S_{v,w'}$  by Lemmas 4.1 and 4.2. Furthermore, now set  $S = S_{v,w'} \cup \{v\}$  is locating-dominating in  $G$ . Indeed,  $w$  separates vertices  $u$  and  $w'$  while  $v$  separates  $w$  from all other vertices of  $G$ . Hence,  $\deg(w') = 3$ .

Assume next that  $z \in N_G(w) \setminus N_G(w')$  and  $z' \in N_G(w') \setminus N_G(w)$ . Consider graph  $G' = G - uw - uw' - vw$ . In this graph  $u$  is a leaf and  $v$  is its support vertex. Now also  $w$  and  $z$  are a leaf and a support vertex, respectively, or  $w$  and  $z$  form a  $P_2$  component. Moreover, there is a possibility that we created a pair of open twins if  $z$  is a support vertex in  $G$ .

Assume first that  $w$  and  $z$  form a  $P_2$  component in  $G'$ . Now  $G'_{w,z} = G' - w - z$  contains fewer closed twins than  $G$ . Thus, there exists an optimal LD-code  $S'$  of  $G'_{w,z}$  that contains  $v$  and  $|S'| \leq \frac{n}{2} - 1$ . Now,  $S = S' \cup \{z\}$  is an LD-code of  $G'$  that contains both  $v$  and  $z$  such that  $|S| \leq \frac{n}{2}$ . Moreover, the vertex  $w'$  is dominated by at least two vertices. When we consider the set  $S$  in  $G$ , the only possible vertices in  $V(G) \setminus S$  which might not be separated are  $u$ ,  $w$  and  $w'$ . However,  $w$  is the only such vertex adjacent to  $z$  and  $w'$  that is dominated by at least two vertices. Hence, either  $w' \in S$  or  $z' \in S$ . In both cases, set  $S$  is locating-dominating in  $G$ .

Assume then that  $z$  is a support vertex in  $G'$  but not in  $G$ . The number of closed twins in  $G'$  is smaller than in  $G$ . Thus, we have an optimal LD-code  $S$  in  $G'$  which contains vertices  $z$  and  $v$  such that  $|S| \leq \frac{n}{2}$ . By the same arguments as in the case above concerning the  $P_2$ -component,  $S$  is locating-dominating also in  $G$ .

Let us next assume that  $z$  is a support vertex and  $\ell_z$  is the adjacent leaf in  $G$ . In this case, we consider subgraph  $G'' = G - uw - vw'$ . Notice that  $G''$  does not contain any open twins and it has a smaller number of closed twins than  $G$ . Hence, it admits an optimal LD-code  $S'$  with  $|S'| \leq \frac{n}{2}$ . By Lemma 4.1 we may assume that  $z \in S'$  and by Lemma 4.2 that  $\ell_z \notin S'$ . Hence, we have  $w \in S'$  or  $v \in S'$  for separating  $\ell_z$  and  $w$ . If  $w \in S'$ , then also set  $S_w = (S' \setminus \{w\}) \cup \{v\}$  is an LD-code of  $G''$ . Moreover,  $S_w$  is an LD-code of  $G$ . Indeed, the only vertices in  $G$  which might not be separated are  $w, w'$  and  $u$ . However,  $w$  is the only such vertex adjacent to  $z$ . If  $w'$  and  $u$  are not separated in  $G$ , then  $w', u \notin S_w$ . However,  $w'$  is dominated by  $S_w$  in  $G''$ , and thus  $z' \in S_w$  and it separates  $w'$  and  $u$  also in  $G$ . Therefore,  $S_w$  is an LD-code of cardinality  $|S_w| \leq \frac{n}{2}$  in  $G$ . Now, the claim follows.  $\square$

### 4.1.3.2 Subcubic graphs with open twins of degree 3

Let  $G$  be any graph,  $F$  be a connected graph and  $U$  be a vertex subset of  $F$ . Then a subgraph  $H$  of  $G$  is called an  $(F; U)$ -subgraph of  $G$  if the following hold.

- (1) There exists an isomorphism  $j : V(F) \rightarrow V(H)$ .
- (2)  $N_G[j(u)] \subseteq V(H)$  and  $j(u)j(v) \in E(H)$  for all  $u \in U$  and  $j(v) \in N_G(j(u))$ .

We note that an  $(F; U)$ -subgraph  $H$  of  $G$  could also be considered as a subgraph of  $G$  isomorphic to  $F$  such that the closed neighborhood in  $G$  of any vertex  $j(u)$  for  $u \in U$  together with the edges in  $G$  incident with  $j(u)$  are also contained in the subgraph  $H$ .

If  $U = \{u_1, u_2, \dots, u_k\}$  and  $H$  is an  $(F; U)$ -subgraph of  $G$ , then we may also refer to  $H$  as an  $(F; u_1, u_2, \dots, u_k)$ -subgraph of  $G$ . We now define a list of pairwise non-isomorphic graphs as follows. In the proof of Theorem 4.7, we show that a connected subcubic graph  $G$  with open twins of degree 3 and at least 7 edges has  $F_0$  and at least one of graphs  $F_i$ ,  $i \in [1, 6]$ , as its subgraph.



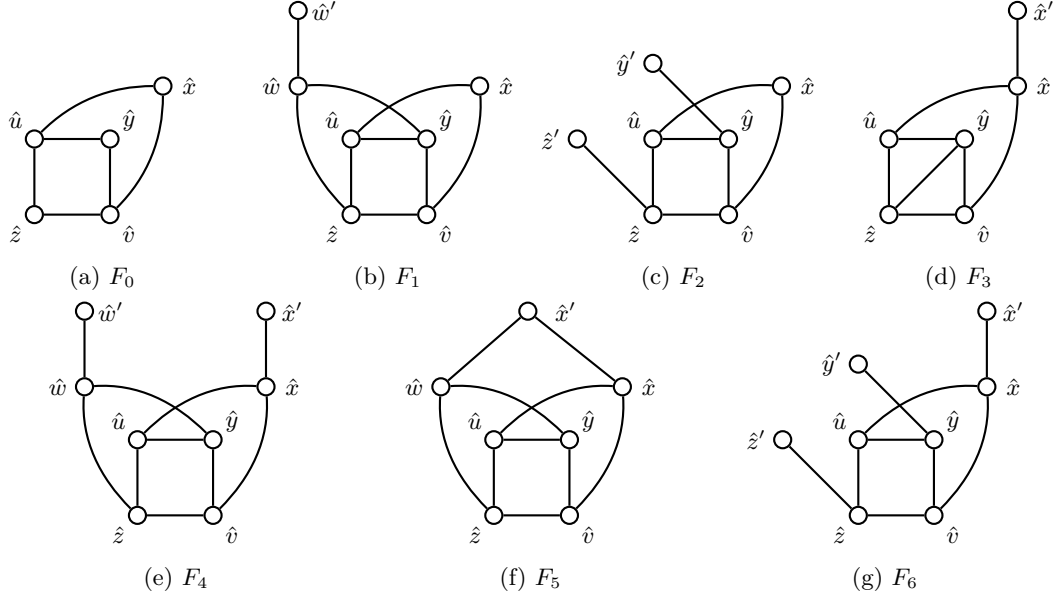


Figure 4.10: A list of pairwise non-isomorphic subcubic graphs. The vertices  $\hat{u}$  and  $\hat{v}$  are open twins of degree 3 in each graph.

- *Graph  $F_0$ :*  $V(F_0) = \{\hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z}\}$  and  $E(F_0) = \{\hat{u}\hat{x}, \hat{u}\hat{y}, \hat{u}\hat{z}, \hat{v}\hat{x}, \hat{v}\hat{y}, \hat{v}\hat{z}\}$ . See Figure 4.10a. The vertices  $\hat{u}$  and  $\hat{v}$  are open twins of degree 3 in  $F_0$ . Any graph that has a pair of open twins of degree 3 has an  $(F_0; \hat{u}, \hat{v})$ -subgraph.
- *Graph  $F_1$ :*  $V(F_1) = \{\hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z}, \hat{w}, \hat{w}'\}$  and  $E(F_1) = \{\hat{u}\hat{x}, \hat{u}\hat{y}, \hat{u}\hat{z}, \hat{v}\hat{x}, \hat{v}\hat{y}, \hat{v}\hat{z}, \hat{w}\hat{u}, \hat{w}\hat{v}, \hat{w}\hat{w}'\}$ . See Figure 4.10b. The pairs  $\hat{u}, \hat{v}$  and  $\hat{y}, \hat{z}$  are open twins of degree 3 in  $F_1$ .
- *Graph  $F_2$ :*  $V(F_2) = \{\hat{u}, \hat{v}, \hat{x}, \hat{x}', \hat{y}, \hat{y}', \hat{z}\}$  and  $E(F_2) = \{\hat{u}\hat{x}, \hat{u}\hat{y}, \hat{u}\hat{z}, \hat{v}\hat{x}, \hat{v}\hat{y}, \hat{v}\hat{z}, \hat{y}\hat{y}', \hat{z}\hat{z}'\}$ . See Figure 4.10c.
- *Graph  $F_3$ :*  $V(F_3) = \{\hat{u}, \hat{v}, \hat{x}, \hat{x}', \hat{y}, \hat{z}\}$  and  $E(F_3) = \{\hat{u}\hat{x}, \hat{u}\hat{y}, \hat{u}\hat{z}, \hat{v}\hat{x}, \hat{v}\hat{y}, \hat{v}\hat{z}, \hat{y}\hat{z}, \hat{x}\hat{x}'\}$ . See Figure 4.10d. The pair  $\hat{u}, \hat{v}$  is open twins of degree 3 in  $F_2$  and the pair  $\hat{y}, \hat{z}$  is closed twins of degree 3 in  $F_2$ .
- *Graph  $F_4$ :*  $V(F_4) = \{\hat{u}, \hat{v}, \hat{w}, \hat{w}', \hat{x}, \hat{x}', \hat{y}, \hat{z}\}$  and  $E(F_4) = \{\hat{u}\hat{x}, \hat{u}\hat{y}, \hat{u}\hat{z}, \hat{v}\hat{x}, \hat{v}\hat{y}, \hat{v}\hat{z}, \hat{w}\hat{u}, \hat{w}\hat{v}, \hat{w}\hat{w}', \hat{x}\hat{x}'\}$ . See Figure 4.10e.
- *Graph  $F_5$ :*  $V(F_5) = \{\hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{x}', \hat{y}, \hat{z}\}$  and  $E(F_5) = \{\hat{u}\hat{x}, \hat{u}\hat{y}, \hat{u}\hat{z}, \hat{v}\hat{x}, \hat{v}\hat{y}, \hat{v}\hat{z}, \hat{w}\hat{u}, \hat{w}\hat{v}, \hat{x}\hat{x}', \hat{w}\hat{x}'\}$ . See Figure 4.10f.
- *Graph  $F_6$ :*  $V(F_6) = \{\hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{x}', \hat{y}, \hat{y}', \hat{z}, \hat{z}'\}$  and  $E(F_6) = \{\hat{u}\hat{x}, \hat{u}\hat{y}, \hat{u}\hat{z}, \hat{v}\hat{x}, \hat{v}\hat{y}, \hat{v}\hat{z}, \hat{x}\hat{x}', \hat{y}\hat{y}', \hat{z}\hat{z}'\}$ . See Figure 4.10g.

Let  $G$  be a graph,  $i \in [0, 6]$  and let  $U_i$  be a vertex subset of  $F_i$ . Moreover, let  $H_i$  be an  $(F_i; U_i)$ -subgraph of  $G$  under a homomorphism  $j_i : V(F_i) \rightarrow V(H_i)$ . Then we fix a certain naming convention for the vertices of  $H_i$  and  $F_i$  as follows. Firstly, we fix a set of 10 symbols, namely  $L = \{u, v, w, w', x, x', y, y', z, z'\}$ . Then, as shown in Figure 4.10, any vertex of  $F_i$  will be denoted by the symbol  $\hat{a} \in L$  for some  $a \in L$ . In addition, any vertex  $j_i(\hat{a})$  of  $H_i$ , for some  $a \in L$ , will be denoted by the symbol  $a$  (that is, by dropping the hat on the symbol  $\hat{a}$ ). We shall call this the *drop-hat naming convention* on  $V(H_i)$ . We hope that the conventions and the namings will become clearer to the reader as we proceed further with their usages.

**Theorem 4.7.** *Let  $G$  be a connected subcubic graph on  $m \geq 7$  edges, not isomorphic to  $K_{3,3}$  and without open twins of degree 1 or 2. Then, we have*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

*Proof.* Let  $n = n(G)$  be the number of vertices and  $m = m(G)$  be the number of edges of the graph  $G$ . Since  $m \geq 7$ , the graph  $G$  is not isomorphic to either  $K_3$  or  $K_4$ . Therefore, if  $G$  does not contain any open twins of degree 3, then the result holds by Theorem 4.6. Hence, we assume from now on that  $G$  has at least one pair of open twins of degree 3. In other words,  $G$  has an  $(F_0; \hat{u}, \hat{v})$ -subgraph. Let  $\mathcal{G}$  denote the set of all connected subcubic graphs without open twins of degrees 1 or 2 and not isomorphic to  $K_{3,3}$ . Notice that in a subcubic graph not isomorphic to  $K_{3,3}$ , for each vertex  $u$  that is a twin of degree 3, there exists exactly one other vertex  $v$  with the same open neighborhood. Let us assume that  $G \in \mathcal{G}$ ,  $\gamma^{\text{LD}}(G) > \frac{n}{2}$  and among those graphs  $G$  has the smallest number of edges  $m \geq 7$ .

We next consider the graph  $F'_3 = F_3 - \{\hat{x}'\} \in \mathcal{G}$  that has one pair of open twins of degree 3, namely  $\hat{u}$  and  $\hat{v}$ . Now, it can be verified that the set  $S'_3 = \{\hat{v}, \hat{z}\}$  is an LD-code of  $F'_3$  with  $|S'_3| < \frac{n}{2}$ . Notice that  $F'_3$  is the smallest graph (with respect to the number of both vertices and edges) with open twins of degree 3 but without open twins of degree 1 or 2 other than  $K_{3,3}$ .

Since, by assumption,  $G$  has an  $(F_0; \hat{u}, \hat{v})$ -subgraph, say  $H_0$ , under an injective homomorphism  $j_0 : V(F_0) \rightarrow V(G)$ , applying the drop-hat naming convention, the vertices  $j_0(\hat{u})$ ,  $j_0(\hat{v})$ ,  $j_0(\hat{x})$ ,  $j_0(\hat{y})$  and  $j_0(\hat{z})$  of  $H_0$  are called  $u$ ,  $v$ ,  $x$ ,  $y$  and  $z$ . Since  $G$  does not have open twins of degree 2, at least two of the vertices in  $\{x, y, z\}$  must have degree 3 in  $G$ . Therefore, without loss of generality, let us assume that  $\deg_G(y) = \deg_G(z) = 3$ . Then, let  $x'$  (if it exists),  $y'$  and  $z'$  be the neighbors of  $x$ ,  $y$  and  $z$ , respectively, in  $V(G) \setminus \{u, v\}$ . Notice that we may possibly have  $y' = z'$ . Then the following two cases arise.

► **Case 1:**  $\deg_G(x) = 2$ .

In this case, if  $y' = z$ , or equivalently,  $z' = y$ , this implies that  $yz \in E(G)$  and therefore,  $y$  and  $z$  are closed twins of degree 3 in  $G$ . Since  $\deg_G(x) = 2$ , this implies that the graph  $G$  is determined on 5 vertices such that  $G \cong F'_3 = F_3 - \{\hat{x}'\}$ . As we have seen,  $\gamma^{\text{LD}}(G) = 2 < \frac{n}{2}$  in this case. Hence this possibility does not arise as it contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . However, the following two other possibilities may arise.

- (1)  $y' = z' = w$ . In this case, if  $\deg_G(w) = 2$  as well, then the graph  $G$  is determined on 6 vertices such that  $G \cong F_1 - \{\hat{w}'\}$ . Now, it can be verified that the set  $S = \{u, z, w\}$  is an LD-code of  $G$  with  $|S| = \frac{n}{2}$ . This contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence, we must have  $\deg_G(w) = 3$ . So, let  $w'$  be the neighbor of  $w$  in  $V(G) \setminus \{y, z\}$ . Now, we cannot have  $w' = x$ , since  $\deg_G(x) = 2$ . This implies that  $G$  has an  $(F_1; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph.
- (2)  $y' \neq z'$ . In this case,  $G$  contains an  $(F_2; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph. ◀

► **Case 2:**  $\deg_G(x) = 3$ .

In this case, if  $x' = y' = z'$ , then the graph  $G$  is isomorphic to  $K_{3,3}$  contradicting our assumption. Hence, this possibility cannot arise. However, any two of  $x'$ ,  $y'$  and  $z'$  could be equal. This implies the following possibilities.

- (1)  $\{x, y, z\} \cap \{x', y', z'\} \neq \emptyset$ . Without loss of generality, let us assume that  $y' = z$ , or equivalently,  $y = z'$ . This implies that  $yz \in E(G)$  and hence, the graph  $G$  has an  $(F_3; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph.
- (2)  $\{x, y, z\} \cap \{x', y', z'\} = \emptyset$  and  $|\{x', y', z'\}| = 2$ . Without loss of generality, let us assume that  $y' = z' = w$ , say. Observe that now  $y$  and  $z$  are open twins of degree 3. Hence, if  $\deg_G(w) = 2$ , then by interchanging the names of the pairs  $w, x$  and by renaming  $x'$  as  $w'$  we end up in  $(F_1; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph of  $G$  as in Case 1. Hence, (in the original notation) we let  $\deg_G(w) = 3$  and let  $w'$  be the neighbor of  $w$  in  $V(G) \setminus \{u, v\}$ . If  $x' = w$ , or equivalently,  $w' = x$ , it implies that  $xw \in E(G)$ . In other words, contrary to our assumption,  $G$  is isomorphic to  $K_{3,3}$ .

Hence, we must have  $\{w, x\} \cap \{w', x'\} = \emptyset$ . Now, if  $x' \neq w'$ , then  $H$  is an  $(F_4; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph; and if  $x' = w'$ , then  $H$  is an  $(F_5; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph. Hence, with this possibility, the graph  $G$  either has an  $(F_4; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph or an  $(F_5; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph.

- (3)  $\{x, y, z\} \cap \{x', y', z'\} = \emptyset$  and  $|\{x', y', z'\}| = 3$ . In this case, there is an  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph in the graph  $G$ .  $\blacktriangleleft$

We prove the theorem by showing in the next claims that none of the above possibilities can arise thus, arriving at a contradiction. First we show that Possibility 1 of Case 1 cannot arise.

**■ Claim 1.**  $G$  has no  $(F_1; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs.

*Proof of claim.* Suppose, to the contrary, that  $G$  contains an  $(F_1; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph, say  $H_1$ . Therefore, by applying the drop-hat naming convention on  $V(H_1)$ , the vertex  $x$  is of degree 2 in  $G$  and the pairs  $u, v$  and  $y, z$  are open twins of degree 3 in  $G$ . Let  $D = \{ux\}$  and let  $G' = G - D$ . This implies that  $G' \in \mathcal{G}$ . Moreover, we have  $8 \leq m(G') < m(G)$  and hence, by the minimality of  $G$ , there exists an LD-code  $S'$  of  $G'$  such that  $|S'| \leq \frac{n}{2}$ . We further notice that the vertex  $x$  is a leaf with its support vertex  $u$  in  $G'$ . Therefore, by Lemmas 4.1 and 4.2, we assume that  $v \in S'$  and  $x \notin S'$ . Now, if  $u \notin S'$ , then  $S'$  is also an LD-code of  $G$ .

Let us, therefore, assume that  $u \in S'$ . If, on the contrary,  $S'$  is not an LD-code of  $G$ , it would mean that, in  $G$ , the vertex  $x$  is not separated by  $S'$  from some other vertex  $p \in N_G(u) \cap N_G(v) \setminus \{S' \cup x'\} \subseteq \{y, z\}$ . Now, since  $y$  and  $z$  are open twins in  $G$  (hence, also in  $G'$ ), it implies that  $S' \cap \{y, z\} \neq \emptyset$ . Without loss of generality, therefore, let us assume that  $y \in S'$  and that the vertices  $x, z$  are not separated by  $S'$  in  $G$ . In this case, we claim that the set  $S = (S' \setminus \{u\}) \cup \{w\}$  is an LD-code of  $G$ . The set  $S$  is a dominating set of  $G$  since each vertex in  $N_G[u]$  has a neighbor in  $\{v, y\} \subset S$ . We therefore show that  $S$  is also a separating set of  $G$ . In particular,  $y$  separates  $u$  from other vertices in  $V(G) \setminus S$ . While  $v$  separates  $x$  and  $z$  from other vertices in  $V(G) \setminus S$  and  $w$  separates  $x$  and  $z$ . Since  $S'$  is an LD-code of  $G'$ , set  $S$  is locating-dominating in  $G$ . Moreover,  $|S| = |S'| \leq \frac{n}{2}$  contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence, this proves that  $G$  cannot contain any  $(F_1; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph.  $\blacksquare$

Next, we show that Possibility 2 of Case 1 cannot arise.

**■ Claim 2.**  $G$  has no  $(F_2; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs.

*Proof of claim.* Suppose, to the contrary, that  $G$  contains an  $(F_2; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph, say  $H_2$ . Applying the drop-hat naming convention on  $V(H_2)$ , the vertex  $x$  has degree 2 in  $G$  and the vertices  $u, v$  are open twins of degree 3 in  $G$ . Now, if  $\deg_G(y') = \deg_G(z') = 1$ , then the graph  $G$  is determined to be isomorphic to  $F_2$  on  $n = 7$  vertices. It can be checked in this case that the set  $S = \{v, y, z\}$  is an LD-code of  $G$  such that  $|S| = 3 < \frac{n}{2}$ . This contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence, we assume that at least one of  $y'$  and  $z'$  has degree of at least 2 in  $G$ . In other words,  $m \geq 9$ . Let  $D = \{ux\}$  and let  $G' = G - D$ . We have  $G' \in \mathcal{G}$ . Moreover,  $8 \leq m(G') < m$  and hence, by the minimality of  $G$ , there exists an LD-code  $S'$  of  $G'$  such that  $|S'| \leq \frac{n}{2}$ . We further notice that the vertex  $x$  is a leaf with its support vertex  $u$  in  $G'$ . Therefore, by Lemmas 4.1 and 4.2, we assume that  $v \in S'$  and  $x \notin S'$ . Now, if  $u \notin S'$ , then  $S'$  is also an LD-code of  $G$ .

Let us assume that  $u \in S'$ . Now, if  $S'$  is not an LD-code of  $G$ , then, in  $G$ , the vertex  $x$  is not separated by  $S'$  from some other vertex  $p \in N_G(u) \cap N_G(v) \setminus (S' \cup \{x\}) \subseteq \{y, z\}$ . Now, in order for  $S'$  to separate the pair  $y, z$  in  $G'$ , we must have  $\{y, y'\} \cap S' \neq \emptyset$  or  $\{z, z'\} \cap S' \neq \emptyset$ . Therefore, without loss of generality, let us assume that  $\{y, y'\} \cap S' \neq \emptyset$ . This implies that  $p = z$ , that is, the vertices  $x, z$  are not separated by  $S'$  in  $G$ . In particular, therefore, we have  $z \notin S'$ . We now claim that the set  $S = (S' \setminus \{u\}) \cup \{z\}$  is an LD-code of  $G$ . The set  $S$  is a dominating set of  $G$  since each vertex in  $N_G[u]$  has a neighbor in  $\{v, z\} \subset S$ . Therefore, we show that  $S$  is also a separating set of  $G$ . First of all,  $I_G(S; u) = \{z\}$  is unique since  $v \in S$  and  $z'$  is dominated by some vertex in  $S'$ . Furthermore,  $v$  separates  $x$  from all other vertices except  $y$ . However, we had  $\{y, y'\} \cap S' \neq \emptyset$ . Thus,  $y \in S$  or  $y$  and  $x$  are separated by  $S$ . Since  $S'$  is locating-dominating in  $G'$ , also all other vertices in  $V(G) \setminus S$  are pairwise separated. Therefore,  $S$  is an LD-code of  $G$  with  $|S| = |S'| \leq \frac{n}{2}$ . This contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence,  $G$  cannot not contain any  $(F_2; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph.  $\blacksquare$

Observe that together Claims 1 and 2 imply that Case 1 above is not possible and in particular  $\deg_G(x) = 3$ . Therefore, graph  $G$  cannot have an  $(F_0; \hat{u}, \hat{v}, \hat{x})$ -subgraph.

■ **Claim 3.**  $G$  has no  $(F_3; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs.

*Proof of claim.* Suppose, to the contrary, that  $G$  has an  $(F_3; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph, say  $H_3$ . Applying the drop-hat naming convention on  $V(H_3)$ , the vertices  $u$  and  $v$  are open twins of degree 3 and  $y$  and  $z$  are closed twins of degree 3 in  $G$ .

We first show that either we can assume  $m \geq 11$ , or else, we end up with a contradiction. Let  $G^* = G - \{u, v, x, y, z\}$ . If  $m \leq 10$ , then  $m(G^*) \leq 2$ . In other words,  $n(G^*) \leq 3$ . If  $n(G^*) = 1$ , that is,  $V(G^*) = \{x'\}$ , then the graph  $G$  is determined on  $n = 6$  vertices to be isomorphic to  $F_3$ . Moreover, it can be verified that the set  $S = \{x', v, z\}$  is an LD-code of  $G$  such that  $|S| = 3 = \frac{n}{2}$ . This contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence, let us now assume that  $n(G^*) = 2$  and that  $V(G^*) = \{x', x''\}$ . Then, we must have  $x'x'' \in E(G)$  and, again, the graph  $G$  is determined on  $n = 7$  vertices. In this case too, it can again be verified that the set  $S = \{x', v, z\}$  is an LD-code of  $G$  such that  $|S| = 3 < \frac{n}{2}$ . This again contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence, we now assume that  $n(G^*) = 3$  and that  $V(G^*) = \{x', x'', x'''\}$ . If  $x'x''' \in E(G)$ , then in order for  $x''$  and  $x'''$  to not be open twins of degree 1 (since  $G$  does not have open twins of degree 1), there must be an edge in  $G^*$  other than  $x'x''$  and  $x'x'''$ . Therefore,  $m(G) \geq 11$  in this case. Thus, let  $x'x''' \notin E(G)$  which implies that  $x''x''' \in E(G)$ . Then the graph  $G$  is determined on  $n = 8$  vertices and it can be verified that the set  $S = \{x', x'', v, z\}$  is an LD-code of  $G$  such that  $|S| = 4 = \frac{n}{2}$ . This again contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence, we may assume that  $m \geq 11$ .

Now, let  $D = \{ux, uz, yz\}$  and  $G' = G - D$ . Then, we have  $G' \in \mathcal{G}$ . Moreover,  $8 \leq m(G') < m$  and hence, by the minimality of  $G$ , there exists an LD-code  $S'$  of  $G'$  such that  $|S'| \leq \frac{n}{2}$ . Notice that the vertices  $u$  and  $z$  are leaves with support vertices  $y$  and  $v$ , respectively, in the graph  $G'$ . Therefore, by Lemmas 4.1 and 4.2, we assume that  $v, y \in S'$  and that  $u, z \notin S'$ . We now claim that  $S'$  is also an LD-code of  $G$ . To prove so, we only need to show that, in the graph  $G$ , the vertices  $u$  and  $z$  are separated by  $S'$  from all vertices in  $V(G) \setminus S'$ . The vertex  $y \in S'$  separates  $u$  and  $z$  from other vertices in  $V(G) \setminus S'$  and vertex  $v \in S'$  separates  $u$  and  $z$  from each other. Hence,  $S'$  is an LD-code of  $G$  with  $|S'| \leq \frac{n}{2}$ . This contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$  and proves that  $G$  has no  $(F_3; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs. ■

Following claim considers Case 2, Possibility 2.

■ **Claim 4.**  $G$  has neither  $(F_4; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ - nor  $(F_5; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs.

*Proof of claim.* Suppose, to the contrary, that  $G$  has an  $(F_4; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph, say  $H_4$ . Then, in the drop-hat naming convention on  $V(H_4)$ , the pairs  $u, v$  and  $y, z$  are open twins of degree 3 in  $G$ . Now, if  $\deg_G(w') = \deg_G(x') = 1$ , then the graph  $G$  is determined on  $n = 8$  vertices to be isomorphic to  $F_4$ . In this case, it can be verified that the set  $S = \{v, w, x, y\}$  is an LD-code of  $G$  with  $|S| = 4 = \frac{n}{2}$ . Hence, we may assume that the degree of  $w'$  or  $x'$  is at least 2 in  $G$ . Therefore, we have  $m \geq 11$ . Now, let  $D = \{ux, uy, wy\}$  and  $G' = G - D$ . Then, we have  $G' \in \mathcal{G}$ . Moreover,  $8 \leq m(G') < m$  and hence, by the minimality of  $G$ , there exists an LD-code  $S'$  of  $G'$  such that  $|S'| \leq \frac{n}{2}$ .

We remark that the following arguments will also be used in the case of  $(F_5; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph, that is when  $w' = x'$ , in the following paragraph. Notice that the vertices  $u$  and  $y$  are leaves with support vertices  $z$  and  $v$ , respectively, in the graph  $G'$ . Hence, by Lemmas 4.1 and 4.2, we assume that  $v, z \in S'$  and that  $u, y \notin S'$ . We now claim that  $S'$  is also an LD-code of  $G$ . To prove so, we only need to show that, in  $G$ , the vertices  $u$  and  $y$  are separated by  $S'$  from all other vertices in  $V(G) \setminus S'$ . First of all,  $z$  separates  $u$  from each other vertex in  $V(G) \setminus S'$  except possibly  $w$ . However, since  $S'$  is locating-dominating in  $G'$ , we have  $\{w, w'\} \cap S' \neq \emptyset$  and  $w'$  separates  $u$  and  $w$ . Similarly,  $v$  separates  $y$  from  $V(G) \setminus S'$  except possibly  $x$ . However, again either  $x \in S'$  or  $x'$  separates  $y$  and  $x$ . This implies that  $S'$  is an LD-code of  $G$  with  $|S'| \leq \frac{n}{2}$ . This contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence,  $G$  cannot have an  $(F_4; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph.

Again, to the contrary, suppose that  $G$  has an  $(F_5; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph, say  $H_5$ . Then, in the drop-hat naming convention on  $V(H_5)$ , the pairs  $u, v$  and  $y, z$  are open twins of degree 3 in  $G$ . Now, if  $\deg_G(x') = 2$ , then the graph  $G$  is determined on  $n = 7$  vertices to be isomorphic to  $F_5$ . In this case, it can be verified that the set  $S = \{v, x', y\}$  is an LD-code of  $G$  with  $|S| = 3 < \frac{n}{2}$ . Hence, we may assume that  $\deg_G(x') = 3$ . Therefore, we have  $m \geq 11$ . Now, again let  $D = \{ux, uy, wy\}$  and  $G' = G - D$ . Then again,  $G' \in \mathcal{G}$ . Moreover,  $8 \leq m(G') < m$  and hence,  $\gamma^{\text{LD}}(G') \leq \frac{n}{2}$ . Notice that in the preceding arguments we did not require the case  $w' = x'$  to be considered. This implies that by the exact same arguments as above, it can be shown that  $G$  cannot have an  $(F_5; \hat{u}, \hat{v}, \hat{w}, \hat{x}, \hat{y}, \hat{z})$ -subgraph.  $\blacksquare$

Finally, we are left only with Possibility 3 of Case 2.

**■ Claim 5.**  $G$  has no  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs.

*Proof of claim.* Suppose, to the contrary, that  $G$  has an  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph, say  $H_6$ . Then, applying the drop-hat naming convention on  $V(H_6)$ , the vertices  $u$  and  $v$  are open twins of degree 3 in  $G$ . Now, if all three of  $x', y'$  and  $z'$  are leaves in  $G$ , then the graph  $G$  is determined on  $n = 8$  vertices and it can be verified that the set  $S = \{u, x, y, z\}$  is an LD-code of  $G$  with  $|S| = \frac{n}{2}$ . This contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Therefore, without loss of generality, let us assume that  $\deg_G(z') \geq 2$  and that  $N_G(z') \setminus \{z\} = \{z''\}$  if  $\deg_G(z') = 2$  and  $N_G(z') \setminus \{z\} = \{z'', z'''\}$  if  $\deg_G(z') = 3$ . In what follows, we simply assume that  $\deg_G(z') = 3$  and that  $N_G(z') \setminus \{z\} = \{z'', z'''\}$ , as the arguments remain intact even when  $\deg_G(z') = 2$  and the vertex  $z'''$  is absent.

► **Case 1: The graph  $G' = G - \{zz'\}$  does not contain open twins of degree 1 or 2.**

In this case, let  $D = \{zz'\}$  and  $G' = G - D$ . Let  $F'_{z'}$  be the component of  $G'$  to which the vertex  $z'$  belongs and let  $F'_z$  be the component of  $G'$  to which the vertex  $z$  belongs. Notice that we may possibly have  $F'_{z'} = F'_z$ . Furthermore, we have  $F'_z, F'_{z'} \in \mathcal{G}$ . Moreover, we have  $8 \leq m(F'_z) < m$  and hence, by the minimality of  $G$ , there exists an LD-code  $S'_z$  of the component  $F'_z$  such that  $|S'_z| \leq \frac{n'_z}{2}$ , where  $n'_z$  is the order of the component  $F'_z$ . For the component  $F'_{z'}$ , we will denote by  $S'_{z'}$  its minimum-ordered LD-code.

►► **Case 1.1:  $F'_{z'} \neq F'_z$ .**

We show that there exists an LD-code  $S'_{z'}$  of the component  $F'_{z'}$  such that  $|S'_{z'}| \leq \frac{n'_{z'}}{2}$ , where  $n'_{z'}$  is the order of the component  $F'_{z'}$ . Let us first assume that  $F'_{z'}$  does not contain any open twins of degree 3. Here we show that  $F'_{z'}$  cannot be isomorphic to either  $K_3$  or  $K_4$ . First of all, we notice that  $F'_{z'}$  is not isomorphic to  $K_4$  since the latter is 3-regular and  $\deg_{F'_{z'}}(z') \leq 2$ . Let us, therefore, assume that  $F'_{z'} \cong K_3$ . Thus, let  $V(F'_{z'}) = \{z', z'', z'''\}$ . Then, we take  $\hat{D}' = \{z'z'', z'z'''\}$  and  $G'' = G - \hat{D}'$ . Then, the vertex  $z'$  is a leaf with support vertex  $z$  in a component, say  $F''$ , of  $G''$  which belongs to  $\mathcal{G}$  and with  $9 \leq m(F'') < m$ . Hence, by the minimality of  $G$ , there exists an LD-code  $S''$  of  $G''$  such that  $|S''| \leq \frac{n}{2} - 1$ . By Lemmas 4.1 and 4.2, we assume that  $z \in S''$  and  $z' \notin S''$ . Then, it can be verified that the set  $S = S'' \cup \{z''\}$  is an LD-code of  $G$  with  $|S| \leq \frac{n}{2}$ . This implies a contradiction to our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence,  $F'_{z'}$  is not isomorphic to  $K_3$  either. This implies, by Theorem 4.6, that there exists an LD-code  $S'_{z'}$  of the component  $F'_{z'}$  such that  $|S'_{z'}| \leq \frac{n'_{z'}}{2}$ . Let us then assume that  $F'_{z'}$  has a pair of open twins of degree 3. Since we have restricted the possible  $(F; U)$ -subgraphs of  $G$  in previous claims, the component  $F'_{z'}$  together with vertex  $z$  and edge  $zz'$  must contain an  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph. This implies that  $8 \leq m(F'_{z'}) < m$  and hence, by the minimality of  $G$ , there exists an LD-code  $S'_{z'}$  of the component  $F'_{z'}$  such that  $|S'_{z'}| \leq \frac{n'_{z'}}{2}$ .

We now claim that the set  $S = S'_{z'} \cup S'_z$  is an LD-code of  $G$ . It can be verified that  $S$  is a dominating set of  $G$ . To prove that  $S$  is also a separating set of  $G$ , we only need to show that the vertex  $z$  is separated by  $S$  from all vertices in  $\{z', z'', z'''\} \setminus S$  and the vertex  $z'$  is separated by  $S$  from the vertices in  $\{u, v\} \setminus S$ . However, the first of these claims is true due to the fact that  $\{u, v\} \cap S'_z \neq \emptyset$  since  $u$  and  $v$  are open twins in the component  $F'_z$ ; and second one holds by the fact that  $\{z', z'', z'''\} \cap S'_{z'} \neq \emptyset$  in order for  $S'_{z'}$  to dominate  $z'$ . Hence,  $S$  is, indeed, an LD-code of  $G$ . Moreover,  $|S| = |S'_{z'}| + |S'_z| \leq \frac{n'_{z'}}{2} + \frac{n'_z}{2} = \frac{n}{2}$  contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . ◀◀

►► **Case 1.2:**  $F'_{z'} = F'_z = G'$ .

Let  $S'_{z'} = S'_z = S'$ . In this case, the set  $S'$  is an LD-code of  $G$  if either both  $z, z' \in S'$  or both  $z, z' \notin S'$ . Since  $u$  and  $v$  are open twins in  $G'$ , it implies that  $\{u, v\} \cap S' \neq \emptyset$ . Therefore, without loss of generality, let us assume that  $v \in S'$ . Let us first assume that  $z \in S'$  and  $z' \notin S'$ . Now, if on the contrary,  $S'$  is not an LD-code of  $G$ , it implies that, the vertex  $z'$  is not separated by  $S$  from a vertex  $p \in N_G(z) \setminus (S' \cup \{z'\}) = \{u\}$  in the graph  $G$ . Therefore, we must have  $\{z'', z'''\} \cap S' \neq \emptyset$  in order for  $S'$  to dominate the vertex  $z'$ . Without loss of generality, let us assume that  $z'' \in S'$ . Since  $z'$  and  $u$  are not separated, we have  $z'' \in \{x, y\}$ . Thus,  $z'$  is adjacent to  $x$  or  $y$ , contradicting the structure implied by  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph. Thus,  $S$  separates the pair  $u, z'$  and is thus an LD-code of  $G$  if  $z \in S'$  and  $z' \notin S'$ .

Let us next assume that  $z' \in S'$  and  $z \notin S'$ . Again, let us assume on the contrary that  $S'$  is not an LD-code of  $G$ . Recall that  $v \in S'$ . This implies that  $S'$  does not separate the vertex  $z$  and another vertex  $p \in (N_G(v) \cap N_G(z')) \setminus (S' \cup \{z\})$ . However, since  $z'$  is not adjacent to  $x$  or  $y$ , we have  $N_G(v) \cap N_G(z') = \{z\}$ . Hence, we cannot select  $p$ . This implies that  $S$  is an LD-code of  $G$  also if  $z' \in S'$  and  $z \notin S'$ . Moreover,  $|S'| \leq \frac{n}{2}$  contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . ◀◀

Hence,  $G$  has no  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs in this case. ◀

► **Case 2: The graph  $G' = G - \{zz'\}$  has open twins of degree 1 or 2.**

In this case, the vertex  $z'$  is an open twin of degree 1 or 2 with some vertex, say  $z^*$ , such that  $N_G(z') \setminus \{z\} = N_G(z^*) \subseteq \{z'', z'''\}$ . Therefore, we have  $1 \leq \deg_G(z^*) \leq 2$ . Let  $D = \{z'z'', z'z'''\}$  and  $G^* = G - D$ . Moreover, let  $F'_{z''}$  be the component of  $G^*$  to which the vertex  $z''$  belongs and let  $F'_z$  be the component of  $G^*$  to which the vertex  $z$  (and also  $z'$ ) belongs. Notice that we may possibly have  $F'_{z''} = F'_z$ . Further notice that  $F'_z, F'_{z''} \in \mathcal{G}$ . Indeed, the only candidates for open twins of degrees 1 or 2 are  $z', z''$  and  $z'''$ . However,  $z'$  is a leaf adjacent to  $z$  which does not have other adjacent leaves. Moreover,  $z^*$  is a neighbor only to the vertices  $z''$  and  $z'''$  in  $F'_{z''}$ . This implies that neither  $z''$  nor  $z'''$  is an open twin with any vertices in  $V(F'_{z''}) \setminus \{z'', z'''\}$ . Furthermore,  $z''$  and  $z'''$  cannot be open twins of degree 1 since then they would be open twins of degree two in  $G$ . Finally, they cannot be open twins of degree 2 in  $G^*$ , since then they would be open twins of degree 3 in  $G$  adjacent to vertex  $z^*$  of degree 2 contradicting Claim 1 or 2. We denote by  $S'_z$  and  $S'_{z''}$  a minimum-ordered LD-code of  $F'_z$  and  $F'_{z''}$ , respectively.

►► **Case 2.1:**  $F'_{z''} \neq F'_z$ .

To begin with, the component  $F'_z$  of  $G^*$  belongs to  $\mathcal{G}$ . Moreover, we have  $9 \leq m(F'_z) < m$  and hence, by the minimality of  $G$ , we have  $|S'_z| \leq \frac{n'_z}{2}$ , where  $n'_z$  is the order of the component  $F'_z$ .

We next show that  $|S'_{z''}| \leq \frac{n'_{z''}}{2}$ , where  $n'_{z''}$  is the order of the component  $F'_{z''}$ . First of all, the component  $F'_{z''} \in \mathcal{G}$ . Let us first assume that  $F'_{z''}$  does not contain any open twins of degree 3. Here we show that  $F'_{z''}$  cannot be isomorphic to either  $K_3$  or  $K_4$ . We notice that  $F'_{z''}$  is not isomorphic to  $K_4$  since the latter is 3-regular and  $\deg_{F'_{z''}}(z'') \leq 2$ . Let us, therefore, assume that  $F'_{z''} \cong K_3$ . Thus,  $V(F'_{z''}) = \{z^*, z'', z'''\}$  and  $z''z''' \in E(F'_{z''})$ . Then, we take  $D' = \{z''z^*, z''z'''\}$  and  $G'' = G - D'$ . Then, the graph  $G'' \in \mathcal{G}$  with  $12 \leq m(G'') < m$  and hence, by the minimality of  $G$ , there exists an LD-code  $S''$  of  $G''$  such that  $|S''| \leq \frac{n}{2}$ . Moreover, notice that the vertices  $z''$  and  $z^*$  are leaves with support vertices  $z'$  and  $z'''$ , respectively, in  $G''$ . Therefore, by Lemmas 4.1 and 4.2, we assume that  $z', z''' \in S''$  and  $z'', z^* \notin S''$ . Then, the set  $S''$  is also an LD-code of  $G$  since  $z''$  is the only vertex in  $V(G) \setminus S''$  with the  $I$ -set  $I_G(S''; z'') = \{z', z'''\}$ . Moreover,  $|S''| \leq \frac{n}{2}$  implies a contradiction to our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . Hence,  $F'_{z''}$  is not isomorphic to  $K_3$  either. This implies, by Theorem 4.6, that there exists an LD-code  $S'_{z''}$  of the component  $F'_{z''}$  such that  $|S'_{z''}| \leq \frac{n'_{z''}}{2}$ .

Let us then assume that  $F'_{z''}$  has a pair of open twins of degree 3. Then by the previous claims together with the fact that neither  $z'', z'''$  nor  $z^*$  can be open twins of degree 3 in  $G^*$ , the component  $F'_{z''}$  must contain an  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraph. This implies that  $8 \leq m(F'_{z''}) < m$  and hence, by the minimality of  $G$ , there exists an LD-code  $S'_{z''}$  of the component  $F'_{z''}$  such that  $|S'_{z''}| \leq \frac{n'_{z''}}{2}$ .

We now claim that the set  $S = S'_{z''} \cup S'_z$  is an LD-code of  $G$ . It can be verified that  $S$  is a dominating set of  $G$ . We notice that  $z'$  is a leaf with support vertex  $z$  in the component  $F'_z$ . Therefore, by

Lemmas 4.1 and 4.2, we have  $z \in S$  and  $z' \notin S$ . Thus, to show that  $S$  is also a separating set of  $G$ , we only need to show that the vertex  $z'$  is separated by  $S$  from all vertices in  $(N_G[z''] \cup N_G[z''']) \setminus S$ . However, this is true due to the fact that  $z \in S$ . Hence,  $S$  is, indeed, an LD-code of  $G$ . Moreover,  $|S| = |S'_{z''}| + |S'_z| \leq \frac{n'_{z''}}{2} + \frac{n'_z}{2} \leq \frac{n}{2}$  contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . ◀◀

►► **Case 2.2:**  $F'_{z'} = F'_z = G^*$ .

Then, let  $S'_{z''} = S'_z = S'$ . By Lemmas 4.1 and 4.2, we have  $z \in S$  and  $z' \notin S$ . Hence, the set  $S'$  is an LD-code of  $G$  if both  $z'', z''' \notin S'$ . Therefore, without loss of generality, let us assume that  $z'' \in S'$ . Now, if on the contrary,  $S'$  is not an LD-code of  $G$ , it implies that the vertex  $z'$  is not separated by  $S$  from a vertex  $p \in (N_G(z'') \cap N_G(z)) \setminus \{z'\} = \emptyset$  in the graph  $G$ . Since  $p$  cannot exist,  $S'$  is an LD-code of  $G$  with  $|S'| \leq \frac{n}{2}$  which contradicts our assumption that  $\gamma^{\text{LD}}(G) > \frac{n}{2}$ . ◀◀

Therefore,  $G$  has no  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs in this case as well. ◀

This proves the claim that  $G$  has no  $(F_6; \hat{u}, \hat{v}, \hat{x}, \hat{y}, \hat{z})$ -subgraphs exhausting all possibilities in Cases 1 and 2. ■

This concludes the proof. □

In particular, our results imply that we may extend Conjecture 2.2 to all cubic graphs with the exception of  $K_4$  and  $K_{3,3}$ .

**Corollary 4.5.** *Let  $G$  be a connected cubic graph other than  $K_4$  or  $K_{3,3}$ . Then,  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$ .*

### 4.1.3.3 Tight examples

In this section, we consider some constructions which show the tightness of our results. First we show that we cannot further relax the twin-conditions in Theorem 4.7 for subcubic graphs.

**Proposition 4.5.** *There exists an infinite subfamily of connected subcubic graphs with*

- (1) *open twins of degree 1;*
- (2) *open twins of degree 2,*

*which have LD-numbers strictly more than half of their order.*

*Proof.* Consider first Claim 1). Let  $G$  be a connected subcubic graph on  $n = 12k$  vertices, for  $k \geq 1$ , as in Figure 4.11. The graph consists of a path  $P_v$  on  $3k$  vertices named  $v_1, \dots, v_{3k}$ . Furthermore, to each vertex  $v_i$  we attach a three vertex path on vertices  $a_i, b_i, c_i$  from the middle vertex  $b_i$ . Let us construct a minimum-ordered LD-code  $S$  for  $G$ . Since each  $a_i$  and  $c_i$  are open twins, we may assume without loss of generality that each  $c_i \in S$ . Furthermore, since each  $a_i$  needs to be dominated, at least one of the vertices  $a_i, b_i \in S$ . If  $b_i \in S$  and  $a_i \notin S$ , then one of the vertices  $v_{i-1}, v_i, v_{i+1}$  is in  $S$  to separate  $v_i$  and  $a_i$ . If  $b_i \notin S$ , then we still require one of the three vertices to dominate  $v_i$ . In other words, a minimum-ordered LD-code contains a dominating set of  $P_v$  which contains at least  $k$  vertices. Hence, we have  $\gamma^{\text{LD}}(G) = |S| \geq k + 6k = \frac{7n}{12}$ . Note that this lower bound is actually tight for this family of graphs as we obtain the value with the shaded vertices in Figure 4.11 which can be verified to form an LD-code.

Consider next Claim 2). Let  $G$  be a connected subcubic graph on  $n = 60k$  vertices, for  $k \geq 1$ , as in Figure 4.12. The graph consists of a path  $P_v$  on  $5k$  vertices named  $v_1, \dots, v_{5k}$  and to each vertex  $v_i$ , we connect an eleven vertex subgraph  $G_i$  as in Figure 4.12. Denote the vertex in  $G_i$  which has an edge to  $v_i$  by  $u_i$ . Figure 4.12 contains a minimum-ordered LD-code of shaded vertices. Notice that each subgraph  $G_i$  has a pair of open twins of degree 2, we need to include one of them in any minimum-ordered LD-code  $S$  of  $G$ . Furthermore, by Lemma 4.1, we may assume that the support vertex belongs to the set  $S$ . However, the support vertex itself is not enough to separate the leaf and the open-twin outside of  $S$ . Hence,  $|S \cap (V(G_i) \setminus \{u_i\})| \geq 6$ . Furthermore, these vertices do not dominate the vertices in the path  $P_v$  which requires  $2k$  vertices in any LD-code (see [186]). Hence, we require at least  $2k$  vertices in the set  $S \cap (V(P_v) \cup \bigcup_{i=1}^{5k} \{u_i\})$ . Therefore, we have  $|S| \geq 6 \cdot 5k + 2k = 32k = \frac{8n}{15} > \frac{n}{2}$ . □

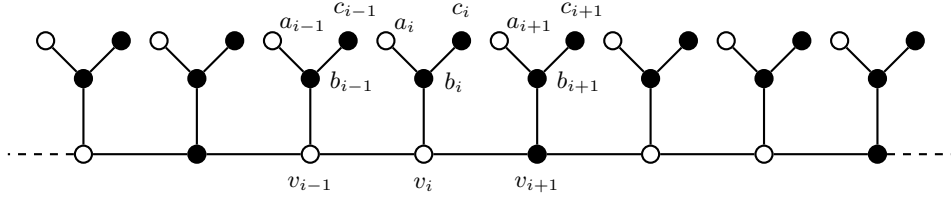


Figure 4.11: Example of a subcubic graph  $G$  on  $n = 12k$  vertices containing open twins of degree 1 and for which  $\gamma^{\text{LD}}(G) = \frac{7}{12}n$ . The shaded vertices constitute a minimum LD-code.

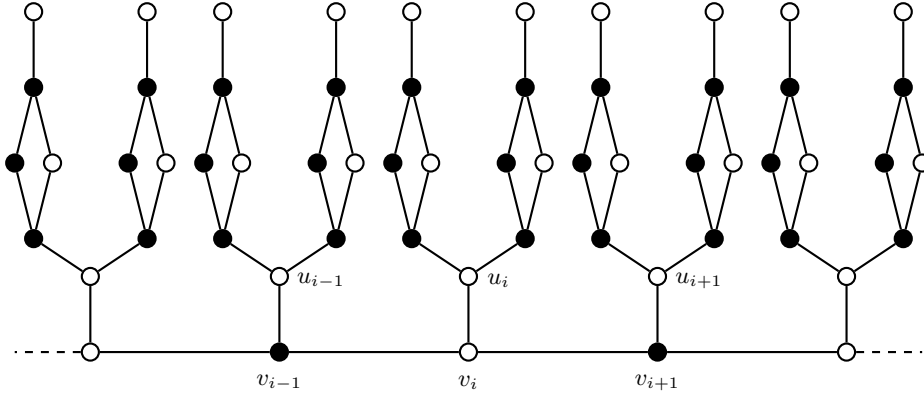


Figure 4.12: Example of a subcubic graph  $G$  on  $n = 60k$  vertices containing open twins of degree 2 and for which  $\gamma^{\text{LD}}(G) = \frac{8}{15}n$ . The shaded vertices constitute a minimum LD-code.

The following proposition shows that the conjecture is not true in general for  $r$ -regular graphs with twins. In other words, the result turns out to be a special property of cubic graphs.

**Proposition 4.6.** *For any integer  $r > 3$ , there exists an infinite subfamily of connected  $r$ -regular graphs with closed twins having location-domination number over half of their order.*

*Proof.* Consider an  $r$ -regular graph  $G_r$  on  $n = (3r + 3)k$  vertices for  $r \geq 4$  as in Figure 4.13. In particular,  $G_r$  contains  $3k$  copies of  $(r-1)$ -vertex cliques (ones within the dashed line in Figure 4.13). Vertices in such cliques are closed twins. We denote these cliques by  $Q_1, \dots, Q_{3k}$ . Furthermore, each clique  $Q_i$  is adjacent to two vertices, let us denote these by  $a_i$  and  $b_i$  so that there is an edge  $a_i b_{i-1}$  for  $2 \leq i \leq 3k$  and  $a_1 b_{3k}$ .

Let us consider a minimum-ordered LD-code  $S$  for  $G_r$ . In particular, we have  $|S \cap Q_i| \geq r - 2$  for each  $i$ . Let us denote by  $c_i \in Q_i$  the single vertex which might not be in  $Q_i \cap S$ . We note that set  $S \cap Q_i$  does not separate vertices  $a_i, b_i, c_i$ . Let us assume that  $|(Q_i \cup \{a_i, b_i\}) \cap S| = r - 2$  for some  $i$ . We observe that then we have  $b_{i-1}, a_{i+1} \in S$ . Hence, we have  $|(Q_{i-1} \cup \{a_{i-1}, b_{i-1}\}) \cup (Q_i \cup \{a_i, b_i\}) \cup (Q_{i+1} \cup \{a_{i+1}, b_{i+1}\}) \cap S| \geq 3r - 4$ . Note that if we had  $|(Q_i \cup \{a_i, b_i\}) \cap S| \geq r - 1$  for each  $i$ , then we would have more vertices in  $S$ . Hence, we have  $|S| \geq \frac{3r-4}{3r+3}n$ . When  $r = 4$  this gives  $\gamma^{\text{LD}}(G) \geq \frac{8}{15}n > \frac{n}{2}$ . We note that this lower bound is attainable with the construction used in Figure 4.13.  $\square$

The following proposition shows that Proposition 4.1 is tight for an infinite subfamily of twin-free subcubic graphs. We remark that there also exists a simpler tight construction of a path which has a leaf attached to all of its vertices. Note that in the case of cubic graphs, no tight constructions are known.

**Proposition 4.7.** *There exists an infinite subfamily of connected twin-free subcubic graphs which have location-domination number equal to half their order.*



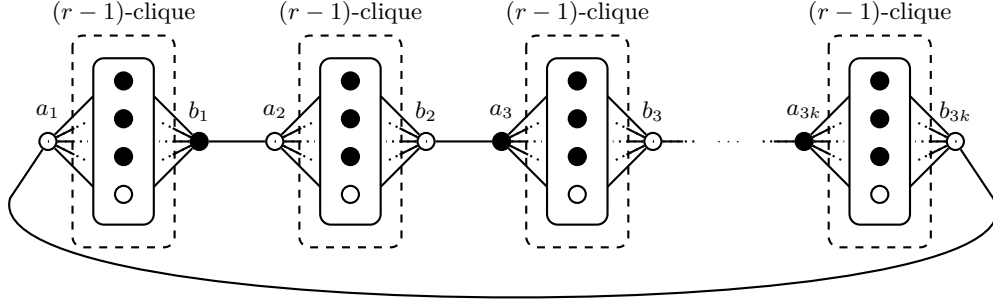


Figure 4.13: Example of an  $r$ -regular graph  $G_r$  on  $n = (3r+3)k$  vertices for which  $\gamma^{\text{LD}}(G_r) = \frac{3r-4}{3r+3}n$ . Therefore, for  $r \geq 4$ , we have  $\gamma^{\text{LD}}(G_r) \geq \frac{8}{15}n$  which proves that Theorem 4.6 is not true for graphs of maximum degree greater than 3. The shaded vertices constitute a minimum LD-code.

*Proof.* Consider the graph  $G$  on  $n = 8k + 2$  vertices as in Figure 4.14. The graph consists of a path on  $4k + 1$  vertices. To each path vertex  $p_i$ ,  $1 \leq i \leq 4k + 1$ , we join a new vertex  $u_i$  and an edge from  $u_i$  to  $u_{i+1}$  if  $i \equiv 2 \pmod 4$  or  $i \equiv 3 \pmod 4$ . Note that in particular  $u_1$  and  $u_{4k+1}$  are leaves. Consider next a minimum-ordered LD-code  $S$  in  $G$ . We note that for each pair  $\{p_i, u_i\}$  where  $u_i$  is a leaf, we have  $\{p_i, u_i\} \cap S \neq \emptyset$  since  $u_i$  is dominated by  $S$ . Let us next show that for each set  $L_j = \{p_{j-1}, p_j, p_{j+1}, u_{j-1}, u_j, u_{j+1}\}$  where the vertices contain a 6-cycle, we have  $|L_j \cap S| \geq 3$ . Suppose, to the contrary, that  $|L_j \cap S| \leq 2$ . Assume first that  $\{p_{j-1}, u_{j-1}\} \cap S = \emptyset$ . To dominate  $u_{j-1}$ , we have  $u_j \in S$ . The only single vertex that can separate both  $p_j$  and  $u_{j+1}$  from  $u_{j-1}$  is  $p_{j+1}$ . However, vertices  $u_j$  and  $p_{j+1}$  cannot separate  $p_j$  and  $u_{j+1}$ . Hence, the assumption  $\{p_{j-1}, u_{j-1}\} \cap S = \emptyset$  leads to  $|L_j \cap S| \geq 3$ , a contradiction. Hence, by symmetry, we assume that  $|\{p_{j-1}, u_{j-1}\} \cap S| = 1$  and  $|\{p_{j+1}, u_{j+1}\} \cap S| = 1$  while  $|\{p_j, u_j\} \cap S| = 0$ . Notice that to dominate both  $u_j$  and  $p_j$  we have  $L_j \cap S = \{p_{j-1}, u_{j+1}\}$  or  $L_j \cap S = \{p_{j+1}, u_{j-1}\}$ . However, the first of these options does not separate  $u_{j-1}$  and  $p_j$  while the second option does not separate  $p_j$  and  $u_{j+1}$ . Hence, we have  $|L_j \cap S| \geq 3$ . This implies that  $\gamma^{\text{LD}}(G) \geq n/2$ . By Proposition 4.1, we have  $\gamma^{\text{LD}}(G) \leq n/2$ . Therefore,  $\gamma^{\text{LD}}(G) = n/2$ .  $\square$

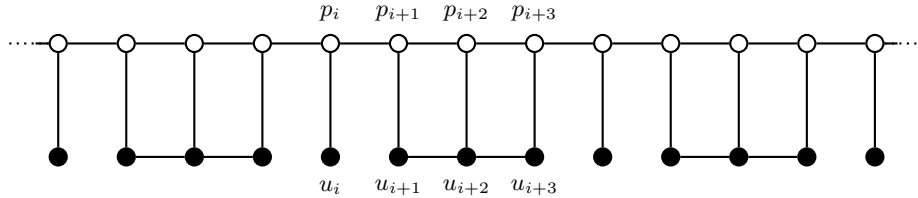


Figure 4.14: Example of a subcubic graph  $G$  on  $n$  vertices for which  $\gamma^{\text{LD}}(G) = \frac{n}{2}$ . The shaded vertices constitute a minimum LD-code.

## 4.2 Location with total domination

In this section, we study on several graph families Conjecture 2.3 which we recall below.

**Conjecture 2.3** ([96]) *Every twin-free and isolate-free graph  $G$  of order  $n$  satisfies*

$$\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n.$$

It is known that any graph of order  $n$  with every component of order at least 3 has a total-dominating set of cardinality at most  $\frac{2}{3}n$  [67], and this bound is tight only for the triangle, the 6-cycle and the family of 2-coronas of graphs [33]. (The 2-corona  $H \circ P_2$  of a connected graph  $H$  is the graph of order  $3|V(H)|$  obtained from  $H$  by attaching a path of length 2 to each vertex of  $H$  so that the resulting paths are vertex-disjoint). However, such a bound does not hold for locating total-dominating codes. For example, subfamilies of (twin-free) graphs with a total-dominating set of cardinality 2 and arbitrarily large LTD-number have been described in [96]. Nevertheless, it seems that in the absence of twins, the LTD-number cannot be as close to the graph's order as it is in the general case. Towards such a fact, in their work in [96], Foucaud and Henning had proposed Conjecture 2.3.

Conjecture 2.3 was proved in [96] for graphs with no 4-cycles as subgraphs. It was also proved for line graphs in [97]. In [96], it was also shown to hold for all graphs with minimum degree at least 26 for which Conjecture 2.2 holds (which is the case, for example, for bipartite graphs and cubic graphs). It was proved in a stronger form for claw-free cubic graphs in [127] (there, the  $\frac{2}{3}$  factor in the upper bound is in fact replaced with  $\frac{1}{2}$ , and the authors conjectured that  $\frac{1}{2}$  holds for all connected cubic graphs, except  $K_4$  and  $K_{3,3}$ ). An approximation of the conjecture was proved to hold for all twin-free graphs in [96], where the  $\frac{2}{3}$  factor in the upper bound is replaced with  $\frac{3}{4}$ . Note that, if true, the bound of Conjecture 2.3 is tight for the 6-cycle and 2-coronas, by the following.

**Observation 4.1.** *If a graph  $G$  of order  $n$  is a triangle, a 6-cycle or a 2-corona of any graph, then*

$$\gamma^{\text{LTD}}(G) = \frac{2}{3}n.$$

In the current work, we give further evidence towards Conjecture 2.3, by showing that it holds for cobipartite graphs (in Section 4.2.1), split graphs (in Section 4.2.2), block graphs (in Section 4.2.3) and subcubic graphs (in Section 4.2.4). Some of our results are actually slightly stronger. Indeed, the proved upper bound for cobipartite graphs is in fact  $\frac{n}{2}$  (which is tight). For twin-free split graphs, we show that the  $\frac{2n}{3}$  bound of the conjecture can never be reached. However, we construct infinitely many connected split graphs that come very close to the bound; this is interesting in its own right showing that it is not just 2-coronas which have such large LTD-numbers. Moreover, the bound for subcubic graphs is proved to hold even if the graphs have twins (except for some small subcubic graphs like  $K_1$ ,  $K_2$ ,  $K_4$  and  $K_{1,3}$ ).

We again refer reader to Table 2.4 for an overview of the literature survey on Conjecture 2.3. The results in this section have also appeared in [39].

### 4.2.1 Cobipartite graphs

We now prove a stronger variant of the bound of Conjecture 2.3, whose proof is a refinement of a similar proof for the (non-total) locating-domination number from [98]. For our next result, we make use of the following result due to Bondy [29] which we recall here.

**Theorem 4.8** (Bondy [29]). *Let  $X$  be a set with  $|X| = k$  and let  $\mathcal{S} = \{X_1, X_2, \dots, X_k\}$  be a collection of  $k$  distinct subsets of  $X$ . Then, there exists an element  $x$  of  $X$  such that  $X_i \setminus \{x\} \neq X_j \setminus \{x\}$  for any two sets  $X_i, X_j \in \mathcal{S}$  and  $i \neq j$ .*

**Theorem 4.9.** *For any twin-free cobipartite graph  $G$  of order  $n$ , we have*

$$\gamma^{\text{LTD}}(G) \leq \frac{n}{2}.$$

*Proof.* Let  $G$  be a twin-free cobipartite graph of order  $n$ . If  $G$  is disconnected, then  $G$  is the disjoint union of two cliques, and thus is not twin-free: it either has closed twins if one of the cliques has order at least 2, or is a pair of open twins if both cliques have order 1, a contradiction. Hence, the graph  $G$  is connected. Let  $C_1$  and  $C_2$  be two cliques of  $G$  that partition its vertex set. Since  $G$  is twin-free, both  $C_1$  and  $C_2$  have cardinality at least 2 where we may assume, renaming  $C_1$  and  $C_2$  if necessary, that  $|C_1| \leq |C_2|$ . Moreover, no two vertices of  $C_1$  have the same neighborhood in  $C_2$ , and vice-versa. Furthermore, at least  $|C_1| - 1$  vertices of  $C_1$  must have neighbors in  $C_2$ , and vice-versa. This implies that both  $C_1$  and  $C_2$  are locating sets of  $G$ . Thus, if any of  $C_1, C_2$  is a total-dominating set, then it is also an LTD-code. Hence, if  $C_1$  is a total-dominating set, we are done, as  $|C_1| \leq \lfloor \frac{n}{2} \rfloor$ . If however  $C_1$  is not a total-dominating set, it means that some vertex  $v$  of  $C_2$  has no neighbors in  $C_1$ ; this vertex is unique since  $G$  is twin-free.

If moreover, there is a vertex  $w$  in  $C_1$  with no neighbor in  $C_2$ , then we select the set  $C = (C_1 \setminus \{w\}) \cup \{x\}$  as a solution set, where  $x \neq v$  is any vertex of  $C_2$  other than  $v$ . This set is clearly a total-dominating set of  $G$ . Moreover, any two vertices of  $C_2$  are located by  $C$ , as  $N_G(v) \cap C = \{x\}$  and any two other vertices of  $C_2 \setminus \{x\}$  have distinct and nonempty neighborhoods in  $C_1$  (and thus, in  $C_1 \setminus \{w\}$ ). Furthermore,  $w$  is the only vertex in  $V(G) \setminus C$  not dominated by  $x$ . Hence,  $C$  is an LTD-code of  $G$ . Since  $|C| = |C_1| \leq \lfloor \frac{n}{2} \rfloor$ , we are done. Therefore, from now on, we assume that every vertex of  $C_1$  has a neighbor in  $C_2$ , more precisely, in the set  $C_v = C_2 \setminus \{v\}$ . Similarly, if  $|C_2| > \lceil \frac{n}{2} \rceil$ , that is, if  $|C_1| < \lfloor \frac{n}{2} \rfloor$ , then by the same preceding arguments, the set  $C_1$  together with any vertex of  $C_2$  other than  $v$  produces an LTD-code of cardinality  $|C_1| + 1 \leq \lfloor \frac{n}{2} \rfloor$ , and we are again done. Hence, we also assume from now on that  $|C_2| = \lceil \frac{n}{2} \rceil$ .

Now, we must have  $|C_2| \geq 3$ , or else, we would have  $|C_1| = |C_2| = 2$  and by our assumption that every vertex in  $C_1$  has a neighbor in  $C_2$  and the fact that the vertex  $v \in C_2$  has no neighbor in  $C_1$ , the two vertices of  $C_1$  must be twins in  $G$ , a contradiction. This implies that both  $C_2$  and  $C_v$  are total-dominating sets of  $G$ . Therefore,  $C_2$  is an LTD-code of  $G$  (recall that  $C_2$  is already a locating set of  $G$ ). Therefore, if  $n$  is even, then we have  $\gamma^{\text{LTD}}(G) \leq |C_2| = \frac{n}{2}$  and we are done. Hence, for the rest of the proof, we assume that  $n$  is odd. Moreover, if  $C_v \setminus N_G(z) \neq \emptyset$  for all  $z \in C_1$ , then we claim that  $C_v$  is an LTD-code of  $G$ . To prove so, since  $C_v$  is a total-dominating set of  $G$ , we only need to prove that  $C_v$  is a locating set of  $G$ . To begin with, all pairs of vertices of  $C_1$  have distinct neighborhoods in  $C_v$  and hence, are located by  $C_v$ . Moreover, with  $N_G(v) \cap C_v = C_v$  and the assumption that  $C_v \setminus N_G(z) \neq \emptyset$  for all  $z \in C_1$ , the vertex  $v$  is located from every vertex  $z$  of  $C_1$ . This proves the claim that  $C_v$  is an LTD-code of  $G$ . We can therefore assume for the rest of the proof that there exists a vertex  $z$  of  $C_1$  with  $C_v \subset N_G(z)$ . Note that there can be at most one such  $z \in C_1$  on account of  $G$  being twin-free. Next, we prove that there exists a vertex  $x \in C_v$  such that the set  $C = C_{vx} \cup \{z\}$  is an LTD-code of  $G$ , where  $C_{vx} = C_v \setminus \{x\}$ .

Recall that any two vertices in  $C_1$  have distinct neighborhoods in  $C_v$ . Moreover, since  $n$  is odd, we have  $|C_v| = |C_1| = \lfloor \frac{n}{2} \rfloor$ . Therefore, in Theorem 4.8, taking  $X = C_v$ ,  $k = \lfloor \frac{n}{2} \rfloor$  and  $\mathcal{S} = \{N_G(u) \cap C_v : u \in C_1\}$  as  $k$  pairwise distinct sets of  $C_v$ , by Theorem 4.8, there exists a subset  $C_{vx}$  of  $C_v$ , for some  $x \in C_v$ , which locates all pairs of vertices of  $C_1$ . In particular, every pair of vertices of  $C_1 \setminus \{z\}$  is located by  $C_{vx}$ . Moreover, since  $N_G(z) \cap C_{vx} = C_{vx}$  (using the assumption that  $N_G(z) \cap C_v = C_v$ ), we therefore have  $C_{vx} \not\subset N_G(u)$  for all  $u \in C_1 \setminus \{z\}$ . This implies that the set  $C_{vx}$  locates both the vertices  $v$  and  $x$  from every vertex  $u$  of  $C_1 \setminus \{z\}$ , since  $C_{vx} \subset N_G(v)$  and  $C_{vx} \subset N_G(x)$ . Finally, since the vertex  $z$  is a neighbor of  $x$  and not of  $v$ , the vertices  $v$  and  $x$  are located by  $z \in C$ . This proves that  $C$  is a locating set of  $G$ . We now show that  $C$  is also a total-dominating set of  $G$ . To prove so, we see that each vertex of  $C_v \cup C_1 \setminus \{z\}$  has  $z \in C$  as its neighbor. Moreover, since  $|C_2| \geq 3$ , that is,  $|C_v| \geq 2$ , the vertices  $v$  and  $z$  have at least one neighbor each in  $C_{vx} \subset C$ . This proves that  $C$  is a total-dominating set of  $G$ . Hence,  $C$  is an LTD-code of  $G$ .

Therefore,  $|C| = |C_v| = \lfloor \frac{n}{2} \rfloor < \frac{n}{2}$  and we are done again. This proves the theorem.  $\square$

The bound of Theorem 4.9 is tight for *complements of half-graphs* (which are graphs with vertex set  $\{x_1, \dots, x_{2k}\}$  and edge set  $\{x_i x_j, |i - j| \leq k - 1\}$ , see [98]). These graphs are cobipartite and have their locating(-total) domination number equal to  $\frac{n}{2}$  [98]. More complicated examples can be found in [98].

## 4.2.2 Split graphs

Consider a split graph  $G = (Q \cup S, E)$  where  $Q$  induces a clique and  $S$  a stable set. We suppose that  $G$  is isolate-free to ensure the existence of an LTD-code in  $G$ , which further implies that  $G$  is connected and  $Q$  non-empty (as every component not containing the clique  $Q$  needs to be an isolated vertex from  $S$ ).

**Theorem 4.10.** *For any twin-free isolate-free split graph  $G = (Q \cup S, E)$  of order  $n$ , we have*

$$\gamma^{\text{LTD}}(G) < \frac{2}{3}n.$$

*Proof.* First, note that we have  $|Q|, |S| \geq 2$  as otherwise  $G$  is a single vertex or not twin-free. Therefore,  $n \geq 4$ . Observe next that  $Q$  is an LTD-code of  $G$  since  $Q$  is a total-dominating set and no two vertices in  $S$  have the same neighbors in  $Q$  (as  $G$  is twin-free) showing that  $Q$  is also locating. Hence, the assertion is true if  $|Q| < \frac{2}{3}n$ , that is, if  $|S| > \frac{1}{3}n$ . Therefore, we can assume henceforth that  $|S| \leq \frac{1}{3}n$ . In particular, we can assume that  $n \geq 6$  because, otherwise, if  $n = 5$ , we would have  $|S| \geq 2 > \frac{1}{3}n$ .

Consider now any set  $D$  consisting of all vertices in  $S$  and, for each  $s \in S$ , some arbitrary neighbor  $q_s \in Q$  (which exists since  $G$  is connected). The set  $D$  is an LTD-code of  $G$  since  $D$  is a total-dominating set and no two vertices in  $Q \setminus D$  have the same neighbors in  $S$  (as  $G$  is twin-free) implying that  $D$  is also locating. Now, we will see how to build such a set  $D$  that is also of the required cardinality. Note that there exist two vertices  $s, s' \in S$  for which  $N(s) \cap N(s') \neq \emptyset$ . This is because, if, on the contrary, for each pair of vertices  $x, y \in S$ , their neighborhoods are disjoint, it implies that

- (1) either the vertices of  $N(x)$  are pairwise twins whenever  $|N(x)| \geq 2$ , a contradiction, or
- (2) each set  $N(x)$  has cardinality exactly one and so, the rest of the  $\frac{1}{3}n$  vertices in  $Q$  have no neighbors in  $S$ . Since  $n \geq 6$ , it implies that there exist twins in  $Q$ , again a contradiction.

Thus, let  $q_{s,s'} \in N(s) \cap N(s')$  be a common neighbor of  $s$  and  $s'$ . This implies that we can assume the vertices  $q_s$  and  $q_{s'}$  to be equal to  $q_{s,s'}$ . This further implies that

$$|D| = |S| + |\{q_x \in Q : x \in S\} - \{q_{s,s'}\}| \leq 2|S| - 1 < \frac{2}{3}n.$$

This proves the result. □

We next show that the bound of Theorem 4.10 cannot be improved for split graphs of orders that are multiples of 3.

**Proposition 4.8.** *For each integer  $k \geq 3$ , there is a connected twin-free split graph  $G_k$  of order  $n = 3k$  and  $\gamma^{\text{LTD}}(G_k) = 2k - 1$ .*

*Proof.* Let  $Q = \{q_1, \dots, q_k\} \cup \{q'_1, \dots, q'_k\}$  be a clique and  $S = \{s_1, \dots, s_k\}$  a stable set, so that  $N(s_i) = \{q_i, q'_i\}$  for  $1 \leq i < k$  and  $N(s_k) = \{q_1, \dots, q_k\}$ . Note that  $q'_k$  has no neighbor in  $S$  and that the sets  $N(s_i)$  are disjoint for  $1 \leq i < k$ . See Figure 4.15 for an illustration.

Let  $C$  be an LTD-code of  $G_k$ . Consider the  $k - 1$  closed neighborhoods  $N[s_i]$  for  $1 \leq i < k$ . If we have  $|N[s_i] \cap C| \geq 2$  for all  $i$  with  $1 \leq i < k$ , then  $|\bigcup_{1 \leq i < k} (N[s_i] \cap C)| \geq 2k - 2$ , and at least one of the remaining vertices  $s_k, q_k, q'_k$  must belong to  $C$ , as otherwise  $N(q_k) \cap C = N(q'_k) \cap C$  would follow, a contradiction. This implies  $|C| \geq 2k - 1$ .

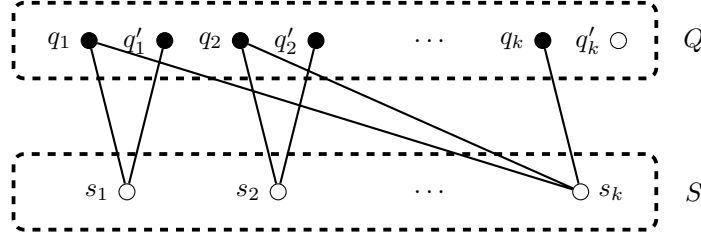


Figure 4.15: The construction of graph  $G_k$  in the proof of Proposition 4.8, with an optimal LTD-code (black vertices).

If, however, for some  $i$  with  $1 \leq i < k$ , we have  $|N[s_i] \cap C| = 1$ , then  $s_i \notin C$  since otherwise  $s_i$  is not totally dominated by the set  $C$ . If  $N[s_i] \cap C = \{q_i\}$ , then  $N(q'_i) \cap C = Q \cap C$ . If  $N[s_i] \cap C = \{q'_i\}$ , then  $N(q_i) \cap C = (Q \cup \{s_k\}) \cap C$ . The two possibilities can occur at most once each. Assume that they both occur once each, with  $N[s_a] \cap C = \{q'_a\}$  and  $N[s_b] \cap C = \{q_b\}$  (with  $1 \leq a < b < k$ ). Note that  $s_k \in C$ , otherwise  $q_a$  and  $q'_b$  are not located. Moreover,  $C$  must contain  $q'_k$  (otherwise  $q'_b$  and  $q'_k$  are not located) and  $q_k$  (otherwise  $q_a$  and  $q_k$  are not located), and so  $|C| \geq 2k - 1$ , as claimed.

Similarly, if we have  $|N[s_i] \cap C| \geq 2$  for all  $i$  with  $1 \leq i < k$  except that  $N[s_a] \cap C = \{q'_a\}$ , if  $s_k \in C$ , then  $q_k \in C$ , otherwise  $q_a$  and  $q_k$  are not located. If  $s_k \notin C$ , then both  $q_k, q'_k$  are in  $C$  to locate the vertices  $q_a, q_k, q'_k$ . Thus, again  $|C| \geq 2k - 1$ .

Finally, if we have  $|N[s_i] \cap C| \geq 2$  for all  $i$  with  $1 \leq i < k$  except that  $N[s_b] \cap C = \{q_b\}$ , if  $s_k \in C$ , then  $q'_k \in C$ , otherwise  $q'_b$  and  $q'_k$  are not located. If  $s_k \notin C$ , then both  $q_k, q'_k$  are in  $C$  to locate the vertices  $q'_b, q_k, q'_k$ , and again  $|C| \geq 2k - 1$ .

Thus, in all the above cases, we have  $|C| \geq 2k - 1$  and, together with the upper bound  $\gamma^{\text{LTD}}(G_k) < \frac{2}{3} \times 3k = 2k$  from Theorem 4.10, we finally obtain  $\gamma^{\text{LTD}}(G_k) = 2k - 1$ .  $\square$

### 4.2.3 Block graphs

We refer the reader to Section 2.1.3 and 4.1.2 for the definitions and notations concerning block graphs. In this section, we also make use of a natural auxiliary tree structure corresponding to a block graph. To that end, we use the following terminology: given any two vertices  $u$  and  $v$  of a tree  $T$ , we say that  $u$  is above  $v$  in  $T$  (or  $v$  is below  $u$  in  $T$ ) if there exists a sequence of vertices  $x_1, x_2, \dots, x_m$  of  $T$  such that, for each  $i = 1, 2, \dots, m - 1$ ,  $x_{i+1}$  is a child of  $x_i$  in  $T$ , where  $m \geq 2$ ,  $x_1 = u$  and  $x_m = v$ .

**Theorem 4.11.** *If  $G \cong P_3$  or if  $G$  is a twin-free and isolate-free block graph of order  $n \geq 4$ , then*

$$\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n.$$

*Proof.* Since the LTD-number of a graph is the sum of the LTD-numbers of each of the components of the graph, it is therefore enough to prove the theorem for a connected twin-free block graph. Thus, let us assume that  $G$  is either isomorphic to a 3-path or is a connected twin-free block graph of order  $n \geq 4$ . The proof is by induction on  $n \geq 3$ . The base case of the induction hypothesis is when  $n = 3$ , in which case  $G \cong P_3$  and  $\gamma^{\text{LTD}}(G) = 2 = \frac{2}{3}n$ . Clearly, any two consecutive vertices of  $P_3$  constitute a minimum LTD-code of  $P_3$  and hence, the result holds for the base case of the induction hypothesis. We now assume, therefore, that  $n \geq 4$  and that the induction hypothesis is true for all connected twin-free block graphs of order at least 3 and at most  $n - 1$ . Next, we construct a new graph  $T_G$  from  $G$  in the following way (see Figure 4.16 for an example of the construction).

For every block  $B$  of  $G$ , introduce a vertex  $u_B \in V(T_G)$  and for every articulation vertex  $c \in V(G)$ , introduce a vertex  $v_c \in V(T_G)$ . Next, we introduce edges  $u_B v_c \in E(T_G)$  if and only if the articulation vertex  $c$  belongs to the block  $B$  of  $G$ . By construction, therefore,  $T_G$  is a tree. Thus, the vertices of the tree  $T_G$  are of two types:

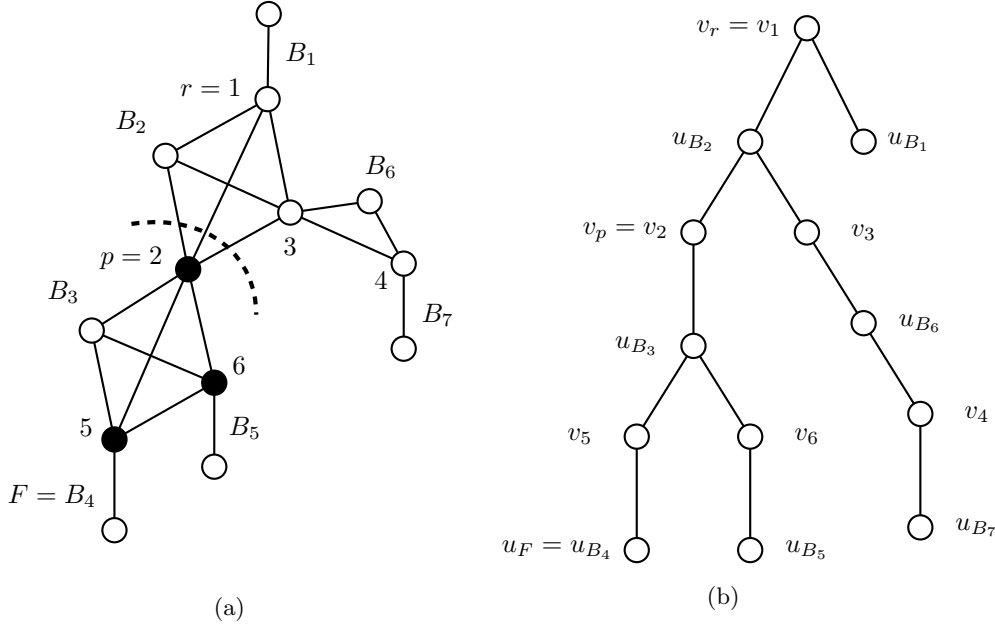


Figure 4.16: Figure (a) represents a twin-free block graph  $G$  and Figure (b) represents  $T_G$ . The vertices underneath the dashed curve represent those deleted from  $G$  to obtain  $G'$ . The black vertices represent vertices in the set  $A$ . (All notations are as in the proof of Theorem 4.11.)

- (1)  $u$ -type:  $u_B$  introduced in a one-to-one association with a block  $B$  of  $G$ ; and
- (2)  $v$ -type:  $v_c$  introduced in a one-to-one association with an articulation vertex  $c$  of  $G$ .

Notice that any pair of vertices  $w, z$  of the tree  $T_G$  such that  $w$  is the grandparent/grandchild of  $z$  in  $T_G$  are of the same vertex type. For a fixed articulation vertex  $r \in V(G)$ , designate  $v_r \in V(T_G)$  as the root of  $T_G$  (indeed, such an articulation vertex exists as  $n \geq 4$  and the twin-free property of  $G$  implies that  $G$  has at least two blocks). Notice that any leaf of the tree  $T_G$  is a vertex of the type  $u_B$  for some leaf block  $B$  of  $G$ . By the twin-free nature of  $G$ , every leaf block  $B$  of  $G$  has order exactly 2. Now, fix a leaf  $u_F$  of  $T_G$  that is at the farthest distance, in  $T_G$ , from the root  $v_r$  of  $T_G$ . We now look at the great-grandparent of the leaf  $u_F$  in the tree  $T_G$  (indeed, the great-grandparent of  $u_F$  in  $T_G$  exists because, on account of  $G$  being twin-free, at least one of the blocks of  $G$  containing the vertex  $r$  has an articulation vertex other than  $r$ ). Notice that the great-grandparent of  $u_F$  in  $T_G$  must be a vertex of the type  $v_p$  for some unique articulation vertex  $p$  of  $G$ . We next define the following.

$$\begin{aligned} \mathcal{B}_p &= \{B : B \text{ is a block of } G \text{ and } u_B \text{ is either a child or a great-grandchild of } v_p \text{ in } T_G\}; \\ U &= \cup_{B \in \mathcal{B}_p} V(B); \text{ and} \\ A &= \{x \in U : x \text{ is an articulation vertex of } G\}. \end{aligned}$$

We now establish the following two claims related to the sets defined above.

■ **Claim 6.** The set  $A$  is an LTD-code of  $G[U]$ .

*Proof of claim.* That  $A$  is a total-dominating set of  $G[U]$  is clear from the structure of  $G$ . We show that  $A$  is also a locating set of  $G[U]$ . So, let us assume that vertices  $w, z \in U \setminus A$ . This implies that both  $w$  and  $z$  are not cut-vertices of  $G$ . This further implies that  $N_{G[U]}(w) \subset A$ , or else, the block of  $G$  that contains  $w$  has another non-articulation vertex which is then a twin with  $w$  in  $G$ , a contradiction. Similarly,  $N_{G[U]}(z) \subset A$ . In other words,  $N_G(w) = N_{G[U]}(w) \cup A$  and  $N_G(z) = N_{G[U]}(z) \cup A$ . Thus, if  $N_{G[U]}(w) \cap A = N_{G[U]}(z) \cap A$ , it implies that  $N_G(w) = N_G(z)$  and thus,  $w$  and  $z$  are twins in  $G$ , again a contradiction. Hence,  $A$  is a locating set of  $G[U]$  and this

proves the claim. ■

■ **Claim 7.**  $|A| \leq \frac{2}{3}|U|$ .

*Proof of claim.* Let  $u_{B_1}, u_{B_2}, \dots, u_{B_m}$  be  $m \geq 1$  children of  $v_p$  in  $T_G$  and let each block  $B_i$  of  $G$  be of order  $n_i$ . Now, due to the twin-free nature of  $G$ , each vertex  $u_{B_i}$  of  $T_G$  has at least  $n_i - 2$  and at most  $n_i - 1$  children (and hence at least  $n_i - 2$  and at most  $n_i - 1$  grandchildren as well). To be more precise, assume that, for  $0 \leq s \leq m$ , the vertices  $u_{B_1}, u_{B_2}, \dots, u_{B_s}$  have exactly  $n_1 - 2, n_2 - 2, \dots, n_s - 2$  children, respectively, in  $T_G$ ; and that the vertices  $u_{B_{s+1}}, u_{B_{s+2}}, \dots, u_{B_m}$  have exactly  $n_{s+1} - 1, n_{s+2} - 1, \dots, n_m - 1$  children, respectively, in  $T_G$ . This implies that we have the following equation.

$$|U| = 1 - s + 2 \sum_{1 \leq i \leq m} (n_i - 1) = 1 - 2m - s + 2 \sum_{1 \leq i \leq m} n_i.$$

Moreover, we have

$$|A| = 1 - s + \sum_{1 \leq i \leq m} (n_i - 1) = 1 - m - s + \sum_{1 \leq i \leq m} n_i.$$

By combining the above two equations, therefore, we have

$$|U| - (2|A| - 1) = s \geq 0 \implies |A| \leq \frac{1}{2}(|U| + 1) \leq \frac{2}{3}|U|,$$

where the last inequality follows from noticing that  $|U| \geq 3$ . Hence, this proves the claim. ■

Now, let  $G' = G - U$ , that is,  $G'$  is the graph obtained by deleting from  $G$  all vertices (and edges incident with them) in the blocks  $B \in \mathcal{B}_p$ . Notice that  $G'$  is still a connected block graph; and assume that the order of  $G'$  is  $n'$  (which is strictly less than  $n$ ). We next divide the proof according to whether  $G'$  is twin-free, has twins or is isomorphic to a 3-path.

► **Case 3:  $G'$  is either twin-free or is isomorphic to a 3-path.**

We further subdivide this case into the following.

►► **Case 3.1:  $n' \leq 2$ .**

In this subcase, if  $n' = 2$ , then the two vertices of  $G'$  form an edge of  $G'$  (since  $G$  is connected). Hence, the two vertices of  $G'$  are closed twins of degree 1, contrary to our initial assumption in this case. Therefore, let us assume that  $n' \leq 1$ . If  $n' = 1$ , then  $v_p$  has no grandparent in  $T_G$ . In other words, there is no vertex of  $v$ -type above  $v_p$  in  $T_G$ . This implies that  $v_p$  must itself be the root vertex of  $T_G$ . However, this, in turn, implies that  $G'$  is an empty graph which contradicts the fact that  $n' = 1$ . Thus, we must have  $n' = 0$ . In this case too,  $v_p$  must itself be the root vertex of  $T_G$  and so,  $G = G[U]$  and  $|U| = n$ . Therefore, by Claim 6, the set  $A$  is an LTD-code of  $G[U] = G$ . Moreover, by Claim 7, we have

$$|A| \leq \frac{2}{3}|U| = \frac{2}{3}n.$$

◀◀

►► **Case 3.2:  $n' \geq 3$ .**

In this subcase, by the induction hypothesis, we have  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n'$ . Suppose now that  $S' \subset V(G')$  is a minimum LTD-code of  $G'$ , that is with  $|S'| = \gamma^{\text{LTD}}(G')$ . We then claim that the set  $S = S' \cup A$  is an LTD-code of  $G$ . To prove so, we first see that the set  $S$  is a total-dominating set of  $G$ , since  $S'$  is a total-dominating set of  $G'$  and  $A$  is a total-dominating set of  $G[U]$ . Moreover,  $S$  is also a locating set of  $G$  due to the following two reasons.

- (1) Any two distinct vertices  $w \in V(G') \setminus S'$  and  $z \in V(G) \setminus S$  are located by  $S'$ .

(2) By Claim 6, the set  $A$  is a locating set of  $G[U]$ .

Using Claim 7, therefore, the two-thirds bound on  $\gamma^{\text{LTD}}(G)$  in this subcase is established by the following inequality.

$$\gamma^{\text{LTD}}(G) \leq |S| = |S'| + |A| \leq \gamma^{\text{LTD}}(G') + \frac{2}{3}|U| \leq \frac{2}{3}(n' + |U|) = \frac{2}{3}n.$$

◀◀

Thus, the result holds in the case that  $G'$  is either twin-free or is isomorphic to a 3-path. ◀

We next turn to the case that  $G'$  has twins.

► **Case 4:  $G'$  is neither twin-free nor isomorphic to a 3-path.**

Assume that  $x$  and  $y$  are two vertices of  $G'$  which are twins in  $G'$ . Then, without loss of generality, there exists an edge in  $G$  between the vertices  $p$  and  $x$  and there is no edge in  $G$  between  $p$  and  $y$ . This implies that  $x$  and  $p$  belong to the same block  $X$ , say, of  $G$  to which  $y$  does not belong. Moreover, notice that there can be only one such  $y$  that is a twin of  $x$  in  $G'$ . Let  $Y$  be a block of  $G$  to which the vertex  $y$  belongs. Next, we prove the following claim.

■ **Claim 8.** If  $x$  does not belong to the block  $Y$ , then  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ .

*Proof of claim.* If  $|V(Y)| \geq 3$ , it would mean that  $x$  would have at least two neighbors, say,  $v$  and  $w$  in  $Y$  (since  $x$  and  $y$  are twins in  $G'$ ). Then the vertices  $v, w, x, y$  would induce a  $K_4$  in the block graph  $G$  making  $x$  belong to  $Y$ , a contradiction. Therefore, we have  $|V(Y)| = 2$ , that is,  $Y$  is a leaf block of  $G$ . So, let  $V(Y) = \{v, y\}$ . Therefore, we have  $v \in N_{G'}(x)$ . Now, assume that  $\deg_G(y) \geq 2$ , that is, there exists some  $w \in N_G(y) \setminus \{v\}$  (with  $w = x$ , possibly). Since  $x$  and  $y$  are twins in  $G'$ , if  $w \neq x$ , we have  $v, w \in N_G(x)$  and thus, the set  $\{v, w, x, y\}$  induces a  $K_4$  in  $G$ . Moreover, if  $w = x$ , then the set  $\{v, x, y\}$  induces a  $K_3$  in  $G$ . Either way, the block  $Y$  with  $|V(Y)| = 2$  is contained inside the subgraph  $K_3$  of  $G$  which is a contradiction, since  $Y$ , being a block, is a maximal complete subgraph of  $G$ . Therefore, we have  $\deg_G(y) = 1$ , that is, the vertex  $y$  is a leaf with  $v$  as its support vertex in  $G$ .

On the other hand, if  $\deg_G(x) \geq 3$ , then  $x$  must have a neighbor  $w$ , say, in  $G$  other than  $p$  and  $v$ . Therefore,  $v, w \in N_{G'}(x)$ . Since,  $x$  and  $y$  are twins in  $G'$ , we must also have  $w \in N_G(y)$  making  $\deg_G(y) \geq 2$ , a contradiction. Hence, we have  $\deg_G(x) = 2$  with  $N_G(x) = \{p, v\}$ . This implies that the set  $\{x, v, y\}$  induces a  $P_3$  in  $G'$ , where, the vertices  $x$  and  $y$  are leaves of  $G'$  with  $v$  as their common support vertex. Let  $X_v$  be the block of  $G$  containing the vertices  $x$  and  $v$ . Notice that  $X_v = X$ , possibly, if  $pv \in E(G)$  (recall that  $X$  is the block of  $G$  containing the vertices  $x$  and  $p$ ). Since  $G' \not\cong P_3$  (by our assumption in this case), the vertex  $v$  must belong to another block of  $G$  other than  $X_v$  and  $Y$ . Now, let  $G'' = G' - x$ . Thus,  $G''$  is still a connected block graph.

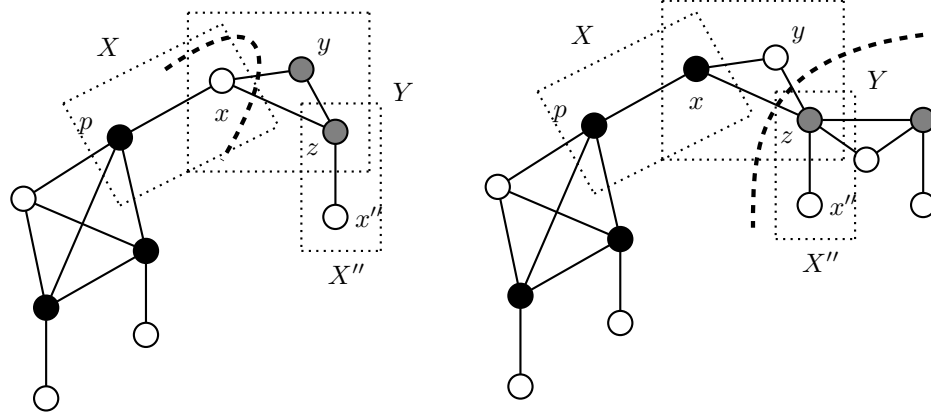
If the graph  $G''$  had twins, then the vertex  $v$  must be one of them. However, this possibility cannot arise as  $y$  is a leaf of  $G''$  with  $v$  as its support vertex. Thus,  $G''$  is also twin-free. Therefore, by our induction hypothesis, let  $S'' \subset V(G'')$  be a minimum LTD-code of  $G''$ . Then, we have  $|S''| = \gamma^{\text{LTD}}(G'') \leq \frac{2}{3}(n' - 1)$ .

We now claim that  $S = A \cup S''$  is an LTD-code of  $G$ . Since, by Claim 6, the set  $A$  is an LTD-code of  $G[U]$  and  $S''$  is assumed to be an LTD-code of  $G''$ , to show that  $S$  is an LTD-code of  $G$ , we have to show that

- (i) the vertex  $x$  is totally dominated and located from every other vertex of  $V(G) \setminus S$  by  $S$ ; and
- (ii) every pair  $s, t$  with  $s \in V(G'') \setminus S''$  and  $t \in U \setminus A$  are located by  $S$ .

To do so, we note that the support vertex  $v$  must belong to the LTD-code  $S''$  of  $G''$ . This implies that the vertex  $x$  is totally dominated by  $v \in S$  and is located from every vertex of  $U \setminus A$  by  $v \in S$ . Moreover, the vertex  $p \in A$ . This implies that  $x$  is located from every vertex of  $V(G'') \setminus S''$  by  $p \in S$ . This implies that  $x$  is located by  $S$  from every vertex of  $V(G) \setminus S$ . Furthermore, let  $s \in V(G'') \setminus S''$





(a) The vertices to the left of the dashed curve represent those deleted from  $G$  to obtain  $G''$ .  $G'' \cong P_3$ . The black vertices constitute the set  $A$  and the gray vertices constitute an LTD-code  $S''$  of  $G''$ .

(b) The vertices to the left of the dashed curve represent those deleted from  $G$  to obtain  $G^*$ .  $G''$  has twins  $x''$  and  $y$ ; and  $G^*$  is a twin-free block graph. The black vertices constitute the set  $A \cup \{x\}$  and the gray vertices constitute an LTD-code  $S^*$  of  $G^*$ .

Figure 4.17: Twin-free block graph  $G$ . The dotted boxes mark the blocks  $X$ ,  $X''$  and  $Y$  of  $G$  as in the proof of Theorem 4.11.

and  $t \in U \setminus A$ . Since the vertex  $t$  of  $U \setminus A$  has a neighbor  $t'$ , say, that is an articulation vertex of  $G$ , it implies that  $t' \in U$ . Hence, we have  $t' \in A$ . This implies that  $t' \in S$  locates the pair  $s, t$ . This proves that  $S$  is an LTD-code of  $G$ . Hence, by Claim 7, we have

$$\gamma^{\text{LTD}}(G) \leq |S| = |S''| + |A| \leq \frac{2}{3}((n' - 1) + |U|) < \frac{2}{3}n.$$

This proves the claim. ■

In view of Claim 8 therefore, we assume for the rest of this proof that  $x$  also belongs to the block  $Y$  of  $G$ . Now, clearly,  $X \neq Y$ , or else,  $py$  would be an edge in  $G$ , contradicting our earlier observation. Thus,  $x$  is an articulation vertex of  $G$  belonging to the distinct blocks  $X$  and  $Y$  of  $G$ . Moreover,  $|V(X)| = 2$ , or else, again,  $x$  and  $y$  would not be twins in  $G'$ , a contradiction. More precisely,  $V(X) = \{x, p\}$ . We also observe here that  $y$  cannot be an articulation vertex of  $G$ , or else,  $x$  and  $y$  would not be twins in  $G'$ , again the contradiction as before. Therefore, since  $G$  is twin-free, every vertex other than  $y$  of the block  $Y$  must be an articulation vertex of  $G$ .

Now, we again look at the block graph  $G'' = G' - x$  on, say,  $n'' (= n' - 1)$  vertices (the graph induced by the vertices on the right of the dashed curve in Figure 4.17a). Notice that, in the tree  $T_G$ , the vertex  $v_x$  cannot have any children other than  $u_X$ , or else,  $x$  and  $y$  cannot be twins in  $G'$ , contrary to our assumption for this case. This implies that the block graph  $G''$  is also connected. We also have  $y \in V(G'')$ . Thus,  $n'' \geq 1$ . However, we can show that  $n'' \neq 2$ . Suppose, to the contrary, that  $n'' = 2$ . In this case, the graph  $G'$  is  $K_3$  which implies that  $G$  has twins, a contradiction. Next, we divide this case into the following subcases according to the order  $n''$  of  $G''$ . ◀◀

►► **Case 4.1:**  $n'' = 1$ .

In this subcase, we claim that  $S = A \cup \{x\}$  is an LTD-code of  $G$ . It is clear that  $S$  is a total-dominating set of  $G$ ; by Claim 6, set  $A$  is an LTD-code of  $G[U]$ . The vertex  $y$  is located from any vertex in  $U \setminus A$  by the vertex  $x$ , and thus the set  $S$  is also locating. Therefore, in this case, using

Claim 7 we have

$$\gamma^{\text{LTD}}(G) \leq |S| = |A| + 1 < \frac{2}{3}(|U| + 2) = \frac{2}{3}n.$$

◀◀

►► **Case 4.2:**  $n'' \geq 3$  and  $G''$  is either twin-free or is isomorphic to a 3-path.

Since  $n''$  is at least 3 and is strictly less than  $n$ , by the induction hypothesis, we have  $\gamma^{\text{LTD}}(G'') \leq \frac{2}{3}n''$ . Moreover, let  $S''$  be a minimum LTD-code of  $G''$ , that is with  $|S''| = \gamma^{\text{LTD}}(G'')$ . We next claim the following.

■ **Claim 9.** The set  $S = S'' \cup A$  is an LTD-code of  $G$ .

*Proof of claim.* Since  $S''$  is a total-dominating set of  $G''$  and  $A$  is a total-dominating set of  $G[U \cup \{x\}]$ , the set  $S$  is therefore a total-dominating set of  $G$ . Next we show that  $S$  is also a locating set of  $G$ . To begin with, we note that  $Y'' = Y - x$  is a block of  $G''$  containing the vertex  $y$ . Now, since  $y$  is not an articulation vertex of  $G$ , we have  $S'' \cap Y'' \neq \emptyset$  (or else,  $y$  is not dominated by  $S''$ ). This implies that  $x$  is located by  $S''$  from all vertices in  $U \setminus A$ . Moreover,  $x$  is also located by  $p$  from all vertices in  $V(G'') \setminus S''$ . Next, any pair  $w, z$  of distinct vertices with  $w \in V(G'') \setminus S''$  and  $z \in V(G) \setminus (S \cup \{x\})$  are located by  $S''$ . Finally, any distinct pair of vertices  $w, z \in U \setminus A$  are located by  $A$ , since the latter is an LTD-code of  $G[U]$  by Claim 6. ■

Therefore, in this subcase, once again by Claim 7, the theorem follows from the following inequality:

$$\gamma^{\text{LTD}}(G) \leq |S| = |S''| + |A| = \gamma^{\text{LTD}}(G'') + |A| \leq \frac{2}{3}(n'' + |U|) < \frac{2}{3}n.$$

◀◀

►► **Case 4.3:**  $n'' \geq 3$  and  $G''$  is neither twin-free nor is isomorphic to a 3-path.

Assume that  $x''$  and  $y''$  are a pair of twins of  $G''$ . Moreover, for  $x''$  and  $y''$  to be twins in  $G''$ , at least one of them must be in the block  $Y$ . Let us, without loss of generality, assume that  $y'' \in V(Y)$ . We next observe that the vertices  $y$  and  $y''$  are the same. To prove so, suppose, to the contrary, that  $y'' \neq y$ . Then  $y''$  is an articulation vertex of  $G$  and so, for  $x''$  and  $y''$  to be twins in  $G''$ ,  $x''$  must not belong to the block  $Y$  of  $G$ . However, this, in turn, implies that  $y$  is a neighbor of  $y''$  but not of  $x''$  and so,  $x''$  and  $y''$  are not twins in  $G''$ , a contradiction all the same. This, therefore, proves the observation.

Again, the vertex  $x'' \notin Y$ , since otherwise,  $x'' \neq y'' = y$  implies that  $x''$  is an articulation vertex of  $G$ , thus forcing  $x''$  and  $y''$  to not be twins, contrary to our supposition. Let  $x''$  belong to the block  $X'' (\neq Y)$  of  $G''$  (and of  $G$ ). We now try to establish the structure of the block  $Y$  of  $G$ . Notice that, by the structure of a block graph, the twins  $x''$  and  $y$  in  $G''$  must have a *single* common neighbor  $z$ , say, in  $G''$  such that  $z$  is an articulation vertex of  $G$  belonging to both the blocks  $Y$  and  $X''$  of  $G$ . Furthermore, if the block  $Y$  contains any vertex of  $G$  other than the vertices  $x, y$  and  $z$ , then  $x''$  and  $y$  are not twins in  $G''$ , a contradiction. Thus, we have  $V(Y) = \{x, y, z\}$ .

Next, to understand the structure of the block  $X''$  of  $G''$ , we see that neither can  $X''$  contain any vertex other than  $z$  and  $x''$ , nor can  $x''$  be an articulation vertex of  $G$ ; or else, we again have the contradiction that  $x''$  and  $y$  are not twins in  $G''$ . Therefore, this implies that  $V(X'') = \{x'', z\}$ , that is,  $X''$  is a leaf block of  $G''$  (and of  $G$ ). See Figure 4.17 for the structure of the blocks  $X''$  and  $Y$ .

With that, we look at the block graph  $G^* = G'' - y$  (the graph induced by the vertices on the right of the dashed curve in Figure 4.17b). Then,  $G^*$  is again a connected graph, since  $y$  is not an articulation vertex of  $G$ . Moreover, the order  $n^*$  of  $G^*$  is at least 2 (since  $x'', z \in V(G^*)$ ). If, however,  $n^* = 2$ , then we have  $V(G'') = \{x'', y, z\}$  and thus,  $G''$  is isomorphic to a 3-path, contrary to our assumption in this subcase. Therefore, we have  $n^* \geq 3$ . We next show the following claim.

■ **Claim 10.** The graph  $G^*$  is twin-free.

*Proof of claim.* Suppose, to the contrary, that the block graph  $G^*$  has a pair of twins. In this case, one of them must be the articulation vertex  $z$  of  $G$ . Let  $x^*$  be the other vertex of  $G^*$  such that  $x^*$  and  $z$  are twins in  $G^*$ . Since  $x''$  is a neighbor of  $z$  alone in  $G^*$ , therefore  $z$  cannot be a twin in  $G^*$  of any vertex other than  $x''$ . In other words,  $x^* = x''$ . However, since  $\deg_{G^*}(x'') = 1$ , we have  $\deg_{G^*}(z) = 1$  and, hence, the graph  $G^*$  is simply the edge  $x''z$  of  $G$ . This however, contradicts the fact that  $n^* \geq 3$ . Hence, this proves that  $G^*$  is twin-free. ■

Since  $n^*$  is at least 3 and is strictly less than  $n$ , by the induction hypothesis, we have  $\gamma^{\text{LTD}}(G^*) \leq \frac{2}{3}n^*$ . Moreover, let  $S^*$  be a minimum LTD-code of  $G^*$ , that is with  $|S^*| = \gamma^{\text{LTD}}(G^*)$ . We next claim the following.

■ **Claim 11.** The set  $S = S^* \cup A \cup \{x\}$  is an LTD-code of  $G$ .

*Proof of claim.* Since  $S^*$  is a total-dominating set of  $G^*$  and  $A \cup \{x\}$  is a total-dominating set of  $G[U \cup \{x, y\}]$ , the set  $S$  is therefore a total-dominating set of  $G$ . Next we show that  $S$  is also a locating set of  $G$ . To begin with, we note that, since  $x''$  is a leaf in  $G^*$ , its support vertex  $z$  must be in the LTD-code  $S^*$  of  $G^*$ . Thus, the vertex  $y$  is located from every other vertex in  $V(G) \setminus S$  by the set  $\{x, z\}$ . Next, any pair  $w_1, w_2$  of distinct vertices with  $w_1 \in V(G^*) \setminus S^*$  and  $w_2 \in V(G) \setminus S$ , respectively, are located by  $S^*$ . Finally, by Claim 6, any pair of distinct vertices  $w_1, w_2 \in U \setminus A$  are located by the set  $A$ . ■

Therefore, again using Claim 7, in this subcase, the theorem follows from the following inequality:

$$\gamma^{\text{LTD}}(G) \leq |S| = |S^*| + |A| + 1 < \frac{2}{3}(n^* + |U| + 2) = \frac{2}{3}n.$$

◀◀

This proves the result in the case that  $G'$  is neither twin-free nor isomorphic to a 3-path. ◀

This completes the proof. □

The “twin-free” condition for block graphs is necessary as, without it, the conjecture does not hold: for example, for  $\Delta$ -stars  $K_{1,\Delta}$  with  $\Delta \geq 3$  (which are block graphs), the LTD-number is  $\Delta$ . On the other hand, for any block graph  $H$  of order  $k \geq 2$ , the 2-corona  $G = H \circ P_2$  is a twin-free block graph of order  $n = 3k$  and by Observation 4.1, it has LTD-number equal to its total domination number, that is,  $\gamma^{\text{LTD}}(G) = \gamma_t(G) = 2k = \frac{2}{3}n$ . See Figure 4.18 for an illustration with  $H$  a complete graph. Thus, we obtain the following.

**Proposition 4.9.** *There are infinitely many connected twin-free block graphs  $G$  of order  $n$  with  $\gamma^{\text{LTD}}(G) = \frac{2}{3}n$ .*

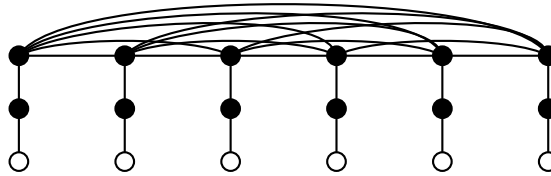


Figure 4.18: The 2-corona  $K_6 \circ P_2$  of a complete graph of order 6.

## 4.2.4 Subcubic graphs

In this section, we establish a tight upper bound on the LTD-number of a subcubic graph, where a subcubic graph is a graph with maximum degree at most 3. For this purpose, let  $\mathcal{F}_{\text{tdom}}$  be the family consisting of the three complete graphs  $K_1$ ,  $K_2$ , and  $K_4$ , and a star  $K_{1,3}$ , that is,

$$\mathcal{F}_{\text{tdom}} = \{K_1, K_2, K_4, K_{1,3}\}.$$

Recall that a *diamond* is the graph  $K_4 - e$  where  $e$  is an arbitrary edge of the  $K_4$ . A *paw* is the graph obtained from a triangle  $K_3$  by adding a new vertex and joining it with an edge to one vertex of the triangle. Equivalently, a paw is obtained from  $K_{1,3}$  by adding an edge between two leaves.

For  $k \geq 2$ , we say a graph  $G$  contains a  $(d_1, d_2, \dots, d_k)$ -sequence if there exists a path  $v_1 v_2 \dots v_k$  such that  $\deg_G(v_i) = d_i$  for all  $i \in [k]$ . We are now in a position to prove the following upper bound on the LTD-number of a subcubic graph.

**Theorem 4.12.** *If  $G \notin \mathcal{F}_{\text{tdom}}$  is a connected subcubic graph of order  $n \geq 3$ , then*

$$\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n.$$

*Proof.* Suppose, to the contrary, that the theorem is false. Among all counterexamples, let  $G$  be one of minimum order  $n$ . If  $n = 3$ , then  $G \cong P_3$  or  $G \cong K_3$ , and in both cases  $\gamma^{\text{LTD}}(G) = 2 = \frac{2}{3}n$ , a contradiction. Hence,  $n \geq 4$ . Suppose  $n = 4$ . By assumption,  $G \notin \{K_4, K_{1,3}\}$ . If  $G$  is a diamond or a paw, then let  $S$  consist of one vertex of degree 2 and one vertex of degree 3, and if  $G$  is a path or a cycle, then let  $S$  consist of two adjacent vertices of degree 2. In all cases,  $S$  is an LTD-code of  $G$  of cardinality 2, and so  $\gamma^{\text{LTD}}(G) \leq 2 < \frac{2}{3}n$ , a contradiction. Hence,  $n \geq 5$ .

Suppose that  $n = 5$ . If  $G$  is a path  $P_5$  or a cycle  $C_5$ , then  $\gamma^{\text{LTD}}(G) = 3 < \frac{2}{3}n$  (choose three consecutive vertices of degree 2), a contradiction. Hence,  $\Delta(G) = 3$ . Let  $v$  be a vertex of degree 3 in  $G$  with neighbors  $v_1, v_2, v_3$ . Let  $v_4$  be the remaining vertex of  $G$ . Since  $G$  is connected, we may assume, renaming vertices if necessary, that  $v_1 v_4$  is an edge. The set  $\{v, v_1, v_2\}$  is an LTD-code of  $G$ , and so  $\gamma^{\text{LTD}}(G) \leq 3 < \frac{2}{3}n$ , a contradiction. Hence,  $n \geq 6$ .

Suppose that  $n = 6$ . If  $G$  is a path  $P_6$  or a cycle  $C_6$ , then  $\gamma^{\text{LTD}}(G) = 4 = \frac{2}{3}n$  (choose four consecutive vertices of degree 2), a contradiction. Hence,  $\Delta(G) = 3$ . Let  $v$  be a vertex of degree 3 in  $G$  with neighbors  $v_1, v_2, v_3$ , and let  $v_4$  and  $v_5$  be the two remaining vertices of  $G$ . Since  $G$  is connected, we may assume, renaming vertices if necessary, that  $v_1 v_4$  is an edge. One of the sets  $\{v, v_1, v_2, v_4\}$  and  $\{v, v_1, v_3, v_4\}$  is an LTD-code of  $G$ , and so  $\gamma^{\text{LTD}}(G) \leq 4 = \frac{2}{3}n$ , a contradiction. Hence,  $n \geq 7$ .

In what follows, we adopt the notation that if there is a  $(d_1, d_2, \dots, d_k)$ -sequence in  $G$ , then  $P: v_1 v_2 \dots v_k$  denotes a path in  $G$  associated with such a sequence, where  $\deg_G(v_i) = d_i$  for all  $i \in [k]$ . Further, we let  $G' = G - V(P)$  and let  $G'$  have order  $n'$ , and so  $n' = n - k$ . Recall that  $n \geq 7$ .

We show firstly that there is no vertex of degree 1.

■ **Claim 12.**  $\delta(G) \geq 2$ .

*Proof of claim.* Suppose, to the contrary, that  $\delta(G) = 1$ . We proceed further with a series of structural properties of the graph  $G$  that show that certain  $(d_1, d_2, \dots, d_k)$ -sequences are forbidden.

■■ **Claim 12.1.** The following properties hold in the graph  $G$ .

- (a) There is no  $(1, 3, 1)$ -sequence.
- (b) There is no  $(1, 2, 2)$ -sequence.
- (c) There is no  $(1, 2, 3, 1)$ -sequence.
- (d) There is no  $(1, 2, 3, 2, 1)$ -sequence.

- (e) There is no  $(1, 2, 3)$ -sequence.
- (f) There is no  $(1, 2)$ -sequence.
- (g) There is no  $(1, 3, 2)$ -sequence.
- (h) There is no  $(1, 3, 3, 1)$ -sequence.

*Proof of claim.* (a) Suppose that there is a  $(1, 3, 1)$ -sequence in  $G$ . In this case,  $n' = n - 3 \geq 4$ . Since  $G$  is connected, so too is the graph  $G'$ . Let  $v'$  be the third neighbor of  $v_2$  in  $G$  not on the path  $P$ . Suppose  $G' \in \mathcal{F}_{\text{tdom}}$ , implying that  $G' \cong K_{1,3}$  with  $v'$  as a leaf in  $G'$ . The graph  $G$  is therefore determined, and has order  $n = 7$ . In this case, choosing  $S$  to consist of the two support vertices (of degree 3) and a leaf neighbor of each support vertex produces an LTD-code of  $G$  of cardinality 4, and so  $\gamma^{\text{LTD}}(G) \leq 4 < \frac{2}{3}n$ . Hence,  $G' \notin \mathcal{F}_{\text{tdom}}$ . Since  $G'$  is not a counterexample, it holds that  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \gamma^{\text{LTD}}(G') + 2 \leq \frac{2}{3}n$ , a contradiction.

(b) Suppose that there is a  $(1, 2, 2)$ -sequence in  $G$ . Let  $v'$  be the second neighbor of  $v_3$ . As in the previous case,  $n' = n - 3 \geq 4$  and  $G'$  is connected. By part (a), there is no  $(1, 3, 1)$ -sequence, implying that  $G' \notin \mathcal{F}_{\text{tdom}}$  and  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . As before every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction.

(c) Suppose that there is a  $(1, 2, 3, 1)$ -sequence in  $G$ . In this case,  $n' = n - 4 \geq 3$  and  $G'$  is connected. By part (a), there is no  $(1, 3, 1)$ -sequence, implying that  $G' \notin \mathcal{F}_{\text{tdom}}$  and  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}(n - 4) < \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction.

(d) Suppose that there is a  $(1, 2, 3, 2, 1)$ -sequence in  $G$ . In this case,  $G'$  is connected and  $n' = n - 5 \geq 2$ . If  $G' \in \mathcal{F}_{\text{tdom}}$ , then  $G' \cong K_2$  by the fact that there is no  $(1, 3, 1)$ -sequence in  $G$  by part (a). The graph  $G$  is therefore determined, and is obtained from a star  $K_{1,3}$  by subdividing every edge once. We note that  $G$  has order  $n = 7$  and the set  $N[v_3]$  (of non-leaves of  $G$ ) is an LTD-code of  $G$ , implying that  $\gamma^{\text{LTD}}(G) \leq 4 < \frac{2}{3}n$ , a contradiction. Hence,  $G' \notin \mathcal{F}_{\text{tdom}}$ . Thus,  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}(n - 5) < \frac{2}{3}n - 3$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2, v_3$ , and  $v_4$ , implying that  $\gamma^{\text{LTD}}(G) < \frac{2}{3}n$ , a contradiction.

(e) Suppose that there is a  $(1, 2, 3)$ -sequence in  $G$ . In this case,  $n' = n - 3 \geq 4$  and  $G'$  contains at most two components. Let  $v_4$  and  $v'_4$  be the two neighbors of  $v_3$  different from  $v_2$ . By our earlier observations, each of  $v_4$  and  $v'_4$  has degree at least 2 in  $G$ , and therefore degree at least 1 in  $G'$ .

Suppose that  $G'$  is disconnected. In this case, since there is no  $(1, 3, 1)$ -sequence, no  $(1, 2, 3, 1)$ -sequence, and no  $(1, 2, 3, 2, 1)$ -sequence in  $G$ , neither component of  $G'$  belongs to  $\mathcal{F}_{\text{tdom}}$ . By linearity, we therefore have that  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction. Hence,  $G'$  is connected. Recall that  $n' \geq 4$ .

Suppose now that  $G'$  is connected. If  $G' \in \mathcal{F}_{\text{tdom}}$ , then  $G' \cong K_{1,3}$ . Let  $v_5$  be the central vertex of  $G'$ , and so each of  $v_4$  and  $v'_4$  is a leaf neighbor of  $v_5$  in  $G'$ . The graph  $G$  is therefore determined and  $n = 7$ . The set  $\{v_2, v_3, v_4, v_5\}$  is an LTD-code of  $G$ , implying that  $\gamma^{\text{LTD}}(G) \leq 4 < \frac{2}{3}n$ , a contradiction. Hence,  $G' \notin \mathcal{F}_{\text{tdom}}$ . Thus,  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction.

(f) Since there is no  $(1, 2, 1)$ -sequence (since  $n \geq 7$ ), no  $(1, 2, 2)$ -sequence by (b) and no  $(1, 2, 3)$ -sequence by (e), there can be no  $(1, 2)$ -sequence in  $G$ . Hence, part (f) follows immediately from parts (b) and (e).

(g) Suppose that there is a  $(1, 3, 2)$ -sequence in  $G$ . In this case,  $n' = n - 3 \geq 4$ . If  $G'$  is disconnected, then by parts (a)–(f), neither component of  $G'$  belongs to  $\mathcal{F}_{\text{tdom}}$ . By linearity, we therefore have that  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction. Hence,  $G'$  is connected. If

$G' \in \mathcal{F}_{\text{tdom}}$ , then  $G' \cong K_{1,3}$ . In this case, the graph  $G$  has order  $n = 7$  and is obtained from a 5-cycle by selecting two non-adjacent vertices on the cycle and adding a pendant edge to these two vertices. In this case, the set consisting of the two vertices of degree 3 and any two vertices of degree 2 is an LTD-code of  $G$ , implying that  $\gamma^{\text{LTD}}(G) \leq 4 < \frac{2}{3}n$ , a contradiction. Hence,  $G' \notin \mathcal{F}_{\text{tdom}}$ . Thus,  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction.

(h) Suppose that there is a  $(1, 3, 3, 1)$ -sequence in  $G$ . In this case,  $n' = n - 4 \geq 3$  and  $G'$  contains at most two components. Let  $u_i$  be the neighbor of  $v_i$  not on  $P$  for  $i \in \{2, 3\}$ . Possibly,  $u_2 = u_3$ . By parts (a) and (g), the vertex  $u_i$  has degree 3 in  $G$  for  $i \in \{2, 3\}$ . Suppose that  $G'$  is disconnected. In this case, by parts (a)–(g), neither component of  $G'$  belongs to  $\mathcal{F}_{\text{tdom}}$ . By linearity, we therefore have that  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}(n - 4) < \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) < \frac{2}{3}n$ , a contradiction. Hence,  $G'$  is connected. Recall that  $n' \geq 3$ . By parts (a)–(g), we note that  $G' \notin \mathcal{F}_{\text{tdom}}$ , implying that  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' < \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) < \frac{2}{3}n$ , a contradiction.

Thus, the proof of the subclaim is complete. ■ ■

We now return to the proof of Claim 12. By Subclaim 12.1(f), the neighbor of every vertex of degree 1 has degree 3 in  $G$ . Further by Subclaim 12.1(a) and (g), such a vertex of degree 3 has both its other two neighbors of degree 3. Therefore the existence of a vertex of degree 1 implies that there is a  $(1, 3, 3)$ -sequence in  $G$ . In this case,  $n' = n - 3 \geq 4$ . Let  $u_2$  be the neighbor of  $v_2$  not on  $P$ , and let  $u_3$  and  $w_3$  be the two neighbors of  $v_3$  not on  $P$ . By our earlier observations, the vertex  $u_2$  has degree 3 in  $G$ , and, by Subclaim 12.1(h), both vertices  $u_3$  and  $w_3$  have degree at least 2 in  $G$ .

Suppose that  $G'$  contains a component that belongs to  $\mathcal{F}_{\text{tdom}}$ . By Subclaim 12.1, this is only possible if  $u_3$  and  $w_3$  are adjacent and both vertices have degree 2 in  $G$ . In this case,  $G[\{v_3, u_3, w_3\}]$  is a triangle in  $G$ . We now consider the connected graph  $G^* = G - \{v_1, v_2, v_3, u_3, w_3\}$  of order  $n^* = n - 5$ . Since  $u_2$  has degree 2 in  $G^*$ , we note that  $n^* \geq 3$  and  $G^* \notin \mathcal{F}_{\text{tdom}}$ . Hence,  $\gamma^{\text{LTD}}(G^*) \leq \frac{2}{3}n^* = \frac{2}{3}(n - 5) < \frac{2}{3}n - 3$ . Every  $\gamma^{\text{LTD}}$ -set of  $G^*$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$ ,  $v_3$ , and  $u_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \gamma^{\text{LTD}}(G^*) + 3 < \frac{2}{3}n$ , a contradiction. Hence, no component of  $G'$  belongs to the family  $\mathcal{F}_{\text{tdom}}$ . By linearity, we therefore have that  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) < \frac{2}{3}n$ , a contradiction. This completes the proof of Claim 12. ■

By Claim 12, every vertex in  $G$  has degree 2 or 3.

■ **Claim 13.** The graph  $G$  is triangle-free.

*Proof of claim.* Suppose that  $G$  contains a triangle  $T$ . Among all triangles in  $G$ , let  $T$  contain the maximum number of vertices of degree 2 in  $G$ . Let  $V(T) = \{v_1, v_2, v_3\}$ , where  $2 \leq \deg_G(v_1) \leq \deg_G(v_2) \leq \deg_G(v_3) \leq 3$ . Since  $n \geq 7$ , the triangle  $T$  contains at most two vertices of degree 2, and so  $\deg_G(v_3) = 3$ . Let  $G' = G - V(T)$  and let  $G'$  have order  $n'$ , and so  $n' = n - 3 \geq 4$ .

Suppose that  $\deg_G(v_1) = 2$ . We note that  $\deg_G(v_2) = 2$  or  $\deg_G(v_2) = 3$ . Since every vertex in  $G$  has degree 2 or 3, no component of  $G'$  belongs to  $\mathcal{F}_{\text{tdom}}$ . Hence by linearity,  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction. Hence,  $\deg_G(v_1) = 3$ , implying that every vertex in  $T$  has degree 3 in  $G$ . Hence by our choice of the triangle  $T$ , no vertex of degree 2 in  $G$  belongs to a triangle.

Let  $u_i$  be the neighbor of  $v_i$  not in the triangle  $T$  for  $i \in [3]$ . We note that the vertices  $u_1$ ,  $u_2$ , and  $u_3$  are not necessarily distinct. Suppose that  $G'$  contains no component that belongs to  $\mathcal{F}_{\text{tdom}}$ . By linearity, this yields  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an

LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction. Hence,  $G'$  contains a component that belongs to  $\mathcal{F}_{\text{tdom}}$ . Since  $n \geq 7$  and no vertex of degree 2 in  $G$  belongs to a triangle, this is only possible if either  $G' \cong K_{1,3}$  or if  $G'$  contains a  $K_2$ -component.

On the one hand, if  $G' \cong K_{1,3}$ , then the three vertices  $u_1, u_2$ , and  $u_3$  are leaves in  $G'$  that are adjacent to a common neighbor (of degree 3) in  $G'$ . In this case, the graph  $G$  is determined and  $n = 7$ , and the set  $V(T) \cup \{u_1\}$  is an LTD-code of  $G$ , implying that  $\gamma^{\text{LTD}}(G) \leq 4 < \frac{2}{3}n$ , a contradiction.

On the other hand, if  $G'$  contains a  $K_2$ -component, then renaming vertices if necessary, we may assume that  $u_1$  and  $u_2$  belong to such a component. We note that  $u_1$  and  $u_2$  both have degree 2 in  $G$ , and  $u_1v_1v_2u_2u_1$  is a 4-cycle in  $G$ . Further we note that in this case,  $G'$  contains two components, where the second component contains the vertex  $u_3$ . We now consider the graph  $G^* = G - \{v_1, v_2, v_3, u_1, u_2\}$ . Let  $G^*$  have order  $n^* = n - 5$ . By the fact that  $\delta(G) \geq 2$  by Claim 12, the graph  $G^* \notin \mathcal{F}_{\text{tdom}}$ , implying that  $\gamma^{\text{LTD}}(G^*) \leq \frac{2}{3}n^* = \frac{2}{3}(n - 5) < \frac{2}{3}n - 3$ . Every  $\gamma^{\text{LTD}}$ -set of  $G^*$  can be extended to an LTD-code of  $G$  by adding to it, for example, the vertices  $u_2, v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) < \frac{2}{3}n$ , a contradiction. ■

By Claim 13, the graph  $G$  is triangle-free. We show next that there is no vertex of degree 2.

■ **Claim 14.** The graph  $G$  is a cubic graph.

*Proof of claim.* Suppose, to the contrary, that  $\delta(G) = 2$ . As before, we obtain a series of structural properties of the graph  $G$  that show that certain  $(d_1, d_2, \dots, d_k)$ -sequences are forbidden. These forbidden sequences will enable us to deduce the desired result of the claim that  $G$  must be a cubic graph.

■■ **Claim 14.1.** The following properties hold in the graph  $G$ .

- (a) There is no  $(2, 2, 2)$ -sequence.
- (b) There is no  $(2, 3, 2)$ -sequence.
- (c) There is no  $(2, 2, 3)$ -sequence.
- (d) There is no  $(2, 2)$ -sequence.
- (e) There is no  $(2, 3, 3)$ -sequence.

*Proof of claim.* (a) Suppose that there is a  $(2, 2, 2)$ -sequence in  $G$ . In this case,  $n' = n - 3 \geq 4$ . Since  $n \geq 7$ ,  $\delta(G) = 2$ , and  $G$  contains no triangle, no component of  $G'$  belongs to  $\mathcal{F}_{\text{tdom}}$ . Hence by linearity,  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n$ , a contradiction.

(b) Suppose that there is a  $(2, 3, 2)$ -sequence in  $G$ . As before,  $n' = n - 3 \geq 4$ . Suppose that  $G'$  contains a component that belongs to  $\mathcal{F}_{\text{tdom}}$ . Since there is no  $(2, 2, 2)$ -sequence and  $n \geq 7$ , and since  $\delta(G) \geq 2$  and  $G$  contains no triangle, this is only possible if  $G' \cong K_{1,3}$ . But then the graph  $G$  is determined and  $n = 7$ , and  $\gamma^{\text{LTD}}(G') = 4 < \frac{2}{3}n$  (by considering the set  $N[v_2]$ ), a contradiction. Hence, no component of  $G'$  belongs to  $\mathcal{F}_{\text{tdom}}$ . By linearity, this yields  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \gamma^{\text{LTD}}(G') + 2 \leq \frac{2}{3}n$ , a contradiction.

(c) Suppose that there is a  $(2, 2, 3)$ -sequence in  $G$ . Since there is no  $(2, 2, 2)$ -sequence and no  $(2, 3, 2)$ -sequence in  $G$ , every vertex different from  $v_2$  that is adjacent to  $v_1$  or  $v_3$  has degree 3 in  $G$ . Together with our earlier observations, the graph  $G'$  therefore cannot contain a component that belongs to  $\mathcal{F}_{\text{tdom}}$ . By linearity, we have  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . As before this yields  $\gamma^{\text{LTD}}(G) \leq \gamma^{\text{LTD}}(G') + 2 \leq \frac{2}{3}n$ , a contradiction.

(d) Since there is no  $(2, 2, 2)$ -sequence and no  $(2, 2, 3)$ -sequence, there can be no  $(2, 2)$ -sequence in  $G$  noting that every vertex has degree 2 or 3.

(e) Suppose that there is a  $(2, 3, 3)$ -sequence in  $G$ . Since there is no  $(2, 2, 2)$ -sequence, no  $(2, 3, 2)$ -sequence, and no  $(2, 2, 3)$ -sequence in  $G$ , the graph  $G'$  cannot contain a component that belongs to

$\mathcal{F}_{\text{tdom}}$ . By linearity, this yields  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \gamma^{\text{LTD}}(G') + 2 \leq \frac{2}{3}n$ , a contradiction.

Thus, the proof of the subclaim is complete. ■ ■

By Subclaim 14.1(d), there is no  $(2, 2)$ -sequence. Hence every vertex of degree 2 has both its neighbors of degree 3. Moreover since there is no  $(2, 3, 2)$ -sequence, every vertex of degree 3 has at most one neighbor of degree 2. But this would imply the existence of a  $(2, 3, 3)$ -sequence, contradicting Subclaim 14.1(e). Therefore, there can be no vertex of degree 2 in  $G$ , that is,  $G$  is a cubic graph. This completes the proof of Claim 14. ■

By Claim 14, the graph  $G$  is a cubic graph. Recall that  $G$  is triangle-free. We now consider a  $(3, 3, 3)$ -sequence. The graph  $G'$  cannot contain a component that belongs to  $\mathcal{F}_{\text{tdom}}$ . By linearity, this yields  $\gamma^{\text{LTD}}(G') \leq \frac{2}{3}n' = \frac{2}{3}n - 2$ . Every  $\gamma^{\text{LTD}}$ -set of  $G'$  can be extended to an LTD-code of  $G$  by adding to it the vertices  $v_2$  and  $v_3$ , implying that  $\gamma^{\text{LTD}}(G) \leq \gamma^{\text{LTD}}(G') + 2 \leq \frac{2}{3}n$ , a contradiction. This completes the proof of Theorem 4.12. □

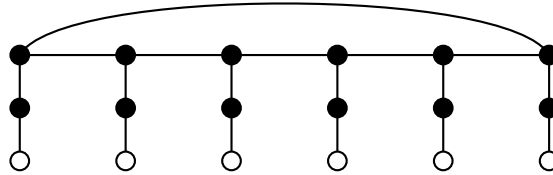


Figure 4.19: The 2-corona  $C_6 \circ P_2$  of a 6-cycle.

For  $k \geq 3$ , the 2-corona  $G = C_k \circ P_2$  of a cycle  $C_k$  has order  $n = 3k$  and by Observation 4.1, it has locating total domination number equal to its total domination number, that is,  $\gamma^{\text{LTD}}(G) = \gamma_t(G) = 2k = \frac{2}{3}n$ . See Figure 4.19 for an illustration. Moreover for  $k \geq 1$ , the 2-corona  $G = P_k \circ P_2$  of a path  $P_k$  has order  $n = 3k$  and also satisfies  $\gamma^{\text{LTD}}(G) = \gamma_t(G) = 2k = \frac{2}{3}n$ . Thus, we obtain the following.

**Proposition 4.10.** *There are infinitely many connected twin-free subcubic graphs  $G$  of order  $n$  with  $\gamma^{\text{LTD}}(G) = \frac{2}{3}n$ .*

## 4.3 Conclusion

The focus of this chapter was to study location in graphs of different families and to prove some of the existing conjectures in the literature. We summarize the work below with respect to the different sections in this chapter and also raise some further questions as future lines of research.

### 4.3.1 LD-codes of block graphs

Block graphs form a subfamily of chordal graphs for which the LD-CODE problem can be solved in linear time [10]. In our study, we complement this result by presenting lower and upper bounds for LD-codes. We gave bounds using both the number of vertices — as it has been done for several other families of graphs — and also using the parameter  $|\mathcal{K}(G)|$  of the number of blocks of  $G$ , that is more fitting for block graphs. In particular, we verified Conjecture 2.2 for block graphs. Moreover,



we addressed the tightness of these bounds to find block graphs where the provided lower and upper bounds are attained.

The structural properties of block graphs have enabled us to prove interesting bounds for their LD-numbers. It would be interesting to see whether other structured families can be studied in a similar way. It would also be interesting to prove Conjecture 2.2 for a larger family of graphs, for example for all chordal graphs. We thus, pose the following open problem.

**Open Problem 4.1.** *Is it true that for all twin-free and isolate-free chordal graphs  $G$  on  $n$  vertices, we have  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$ ?*

### 4.3.2 LD-codes of subcubic graphs

In our study of LD-codes of subcubic graphs, we have proven Conjecture 2.2 for subcubic graphs and answered positively to Problems 4.1 and 4.2. In particular, we showed that for each connected subcubic graph  $G$ , other than  $K_1, K_4, K_{3,3}$  and without open twins of degrees 1 or 2, we have  $\gamma^{\text{LD}}(G) \leq \frac{n}{2}$ . We have also shown that these restrictions on open twins are necessary and that a similar relaxation of conditions in Conjecture 2.2 is not possible for  $r$ -regular graphs in general.

Furthermore, we have presented an infinite family of twin-free subcubic graphs for which this bound is tight. However, the only known tight examples for the  $\frac{n}{2}$ -upper bound over connected twin-free cubic graphs are on six and eight vertices. Moreover, we were unable to find any such tight example for the  $\frac{n}{2}$ -upper bound by manually going through all connected twin-free cubic graphs on ten vertices using the online graph repository [70]. On the other hand, the 10-vertex graph in Figure 4.20 is an example of a cubic graph containing both open and closed twins for which the conjectured upper bound is tight. In [95], Foucaud and Henning asked to characterize every twin-free cubic graph which attains the  $\frac{n}{2}$ -upper bound. We present a new open problem in the same vein:

**Open Problem 4.2.** *Does there exist an infinite family of connected (twin-free) cubic graphs which have LD-number equal to half their order?*

Considering the previous open problem is interesting for both twin-free cubic graphs and graphs which allow twins. It would even be interesting if one could find a single connected twin-free cubic on at least ten vertices which has LD-number equal to half of its order. If there does not exist any such (twin-free) cubic graphs, that also prompts another open problem:

**Open Problem 4.3.** *What is the (asymptotically) tight upper bound for the LD-number of connected (twin-free) cubic graphs on at least ten vertices?*

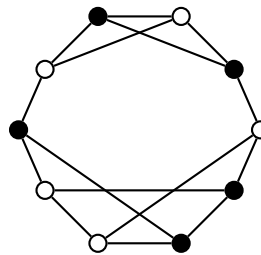


Figure 4.20: Example of a cubic graph  $G$  on  $n = 10$  vertices containing both open and closed twins for which  $\gamma^{\text{LD}}(G) = \frac{n}{2}$ . The shaded vertices constitute a minimum LD-code.

### 4.3.3 Two-thirds conjecture for LTD-codes

As far as our work on LTD-codes is concerned, we have proved Conjecture 2.3 for several important graph families: cobipartite graphs, split graphs, block graphs and subcubic graphs. It would be

interesting to extend these results to larger families, for example chordal graphs (which include split graphs and block graphs). Another interesting subfamily of chordal graphs to consider is the family of interval graphs. It would also be interesting to prove that the bound  $\gamma^{\text{LTD}}(G) \leq \frac{n}{2}$  holds for sufficiently large (twin-free) connected cubic graphs, as conjectured in [127]. We pose these questions in the following.

**Open Problem 4.4.** *Is it true that there exists a positive integer  $n_0$  such that for all twin-free and isolate-free chordal graphs  $G$  on  $n \geq n_0$  vertices, we have  $\gamma^{\text{LTD}}(G) \leq \frac{2n}{3}$ ?*

**Open Problem 4.5.** *Is it true that there exists a positive integer  $n_0$  such that for all twin-free and isolate-free interval graphs  $G$  on  $n \geq n_0$  vertices, we have  $\gamma^{\text{LTD}}(G) \leq \frac{2n}{3}$ ?*

**Open Problem 4.6.** *Is it true that there exists a positive integer  $n_0$  such that for all twin-free and connected cubic graphs  $G$  on  $n \geq n_0$  vertices, we have  $\gamma^{\text{LTD}}(G) \leq \frac{n}{2}$ ?*



## Chapter 5

# Closed separation in graphs

This chapter is dedicated to the study of identifying codes which have the properties of closed separation and domination. As a result, all graphs in this chapter whose ID-codes we look for, are always assumed to be ID-admissible, that is, closed-twin-free. Moreover, since we do not deal with any other codes in this chapter, for a fluency in writing, we interchangeably call an ID-admissible graph to be an *identifiable* graph — the latter terminology has also been used in the literature. In addition, by the word “separation” we shall mean “closed separation” throughout this chapter.

To recall some of the prevalent terminologies around identifying codes, note that, given any code  $C$  of  $G$ , a code vertex in  $C$  is also called a *codeword*. Also recall that a vertex of a code  $C$  that separates a pair of distinct vertices  $u, v \in V(G)$  is called a *separating  $C$ -codeword* for the pair  $u, v$  in  $G$ . In addition, given a vertex  $v$  and a vertex subset  $C$  of a graph  $G$ , the intersection  $N_G[v] \cap C$ , denoted by  $I(v) = I_G(C; v)$ , is called the  $I$ -set of  $v$  with respect to  $C$ .

In this chapter, we study identifying codes of graphs in some special graph families. In Section 5.1, we consider ID-codes of block graphs and study both upper and lower bounds on the ID-numbers of graphs of this class. The upper bound happens to be the following conjecture made by Argiroffo et al. in [7] in the context of block graphs and we thus prove it to be correct.

**Conjecture 5.1** ([7]). *The ID-number of a closed-twin-free block graph is bounded above by the number of blocks in the graph.*

The lower bound on the ID-numbers of the block graphs that we prove are both in terms of the order of the graph and the number of blocks. We also show that all of our bounds on ID-numbers of block graphs are tight by providing examples where these bounds are attained.

In Section 5.2, we study an upper bound conjecture made by Foucaud et al. [99] on identifying codes of graphs in terms of the maximum degree of the graph. The conjecture says the following.

**Conjecture 5.2** ([99]). *There exists a constant  $c$  such that for every connected identifiable graph of order  $n \geq 2$  and of maximum degree  $\Delta \geq 2$ ,*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n + c.$$

We first study Conjecture 5.2 for trees in Section 5.2.2 and then subsequently, in Section 5.2.3, we build on our result for trees and prove the conjecture for all triangle-free graphs. In Section 5.2.2, while proving the conjecture for trees, we also characterize every tree whose ID-number requires a positive constant  $c$  and we also find the exact value of such a positive constant. It turns out that even for triangle-free graphs in Section 5.2.3, the graphs whose ID-numbers require a positive constant  $c$  in Conjecture 5.2 are exactly the trees characterized in Section 5.2.2 which also required a positive constant. As to the graphs whose ID-numbers require such positive constants in Conjecture 5.2, for  $\Delta = 2$  (where the graph is a path or a cycle), it has been long known that  $c = 3/2$  suffices. For each  $\Delta \geq 3$ , we show that  $c = 1/\Delta \leq 1/3$  suffices and that  $c$  is required to have a positive value only for

a finite number of trees. In particular, for  $\Delta = 3$ , there are 12 trees with a positive constant  $c$  and, for each  $\Delta \geq 4$ , the only tree with positive constant  $c$  is the  $\Delta$ -star.

The proof for trees in Section 5.2.2 is based on induction and utilizes results from [100] by Foucaud and Lehtilä. We remark that there are infinitely many trees for which the bound is tight when  $\Delta = 3$ ; for every  $\Delta \geq 4$ , we construct an infinite family of trees of order  $n$  with identification number very close to the bound, namely  $\left(\frac{\Delta-1+\frac{1}{\Delta-2}}{\Delta+\frac{1}{\Delta-2}}\right)n > \left(\frac{\Delta-1}{\Delta}\right)n - \frac{n}{\Delta^2}$ . Furthermore, we also give a new tight upper bound for identification number on trees by showing that the sum of the domination and identification numbers of any tree  $T$  is at most its number of vertices. Our proof for the triangle-free graphs in Section 5.2.3 is also based on induction, whose starting point is our result for trees from Section 5.2.2. Along the way, we prove a generalized version of Bondy's theorem [29] on induced subsets that we use as a tool in our proofs. We also use our main result for triangle-free graphs, to prove the upper bound  $\left(\frac{\Delta-1}{\Delta}\right)n + 1/\Delta + 4t$  for graphs that can be made triangle-free by the removal of  $t$  edges.

Now, using the various definitions related to closed-separating sets, we also recall here (all in one place) the equivalent conditions to imply when a vertex subset of a closed-separable graph  $G$  is a closed-separating set of  $G$ .

**Remark 5.1.** *Let  $G$  be a closed-separable graph and let  $S$  be a vertex subset of  $G$ . Then, the following assertions are equivalent.*

- (1)  $S$  is a closed-separating set of  $G$ .
- (2) Each vertex of  $G$  has a unique closed neighborhood in  $S$ , that is, it has a unique  $I$ -set with respect to  $S$ .
- (3) For all distinct  $u, v \in V(G)$ , we have  $N_G[u] \cap S \neq N_G[v] \cap S$ , that is,  $I_G(S; u) \neq I_G(S; v)$ .
- (4)  $S$  has non-empty intersection with  $\Delta_G(G; u, v) = N_G[u] \Delta N_G[v]$  for all distinct  $u, v \in V(G)$ .

The following remark (also shown in [11]) shows that in order to check if a dominating set  $S$  of a graph  $G$  is an ID-code of  $G$ , we do not need to check if  $S$  separates every pair of distinct vertices of  $G$  but only those which are at a distance of at most 2 between them.

**Remark 5.2.** *Let  $G$  be an identifiable graph. A dominating set  $C$  of  $G$  is an identifying code of  $G$  if and only if  $C$  separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \leq 2$ .*

*Proof.* The necessary condition for the statement follows immediately from the definition of an identifying code. We, therefore, prove the sufficient condition. Thus, it is enough to show that  $C$  separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . So, assume  $u$  and  $v$  to be a pair of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . Since  $C$  is a dominating set of  $G$ , either  $u \in C$  or  $u$  has a neighbor  $w$ , say, in  $C$ . In either case, both  $u$  and  $w$  are not neighbors of  $v$  (since  $d_G(u, v) \geq 3$ ) and hence, at least one of  $u$  or  $w$  must be a separating  $C$ -codeword of the pair  $u, v$  in  $G$ . This proves the result.  $\square$

## 5.1 Block graphs

In this section, we first prove Conjecture 5.1 as a tight upper bound for the ID-numbers of block graphs. We then also prove two tight lower bounds on ID-numbers of block graphs, one, in terms of the number of vertices of the graph and, two, in terms of the number of blocks of the graph. We refer the reader to Section 2.1.3 for the basic definitions and notations concerning block graphs.

All results in this section have also appeared in [47].

### 5.1.1 Upper bounds on ID-numbers of block graphs

In this section, we establish Conjecture 5.1, restated here, on the upper bound on the ID-number of block graphs.

**Conjecture 5.1** ([7]). *The ID-number of a closed-twin-free block graph is bounded above by the number of blocks in the graph.*

The blocks of a block graph provide the graph with a tree-like structure (see, for example, Figure 2.7 in Section 2.1.3 or Figure 4.16 for the auxiliary tree structure of a block graph). Thus, Conjecture 5.1 is in terms of a parameter relevant to block graphs from their structural point of view. Besides proving the above conjecture, we also show that this upper bound is, in fact, tight by providing examples of arbitrarily large-ordered block graphs whose ID-numbers attain the bound. However, we first prove Conjecture 5.1 in the following theorem.

**Theorem 5.1.** *Let  $G$  be a connected closed-twin-free block graph and let  $\mathcal{K}(G)$  be the set of all blocks of  $G$ . Then, we have*

$$\gamma^{\text{ID}}(G) \leq |\mathcal{K}(G)|.$$

*Proof.* Suppose, to the contrary, that there is a closed-twin-free block graph  $G$  of minimum order such that  $\gamma^{\text{ID}}(G) > |\mathcal{K}(G)|$ . We also assume that  $G$  has at least four vertices since it can be easily checked that the theorem is true for closed-twin-free block graphs with at most three vertices (which are only  $P_1$  and  $P_3$ ). Suppose that  $K \in \mathcal{K}(G)$  is a leaf-block of  $G$ . Due to the closed-twin-free property of  $G$ , one can assume that  $V(K) = \{x, y\}$  and, without loss of generality, that  $x$  and  $y$  are the non-articulation and the negative articulation vertices, respectively, of  $K$ . Let  $G' = G - x$  be the graph obtained by deleting the vertex  $x \in V(G)$  (and the edge incident with  $x$ ) from  $G$ . Then  $G'$  is a block graph with  $|\mathcal{K}(G')| = |\mathcal{K}(G)| - 1$ . We now consider the following two cases.

► **Case 5:  $G'$  is closed-twin-free.**

By the minimality of the order of  $G$ , there is an ID-code  $C'$  of  $G'$  such that  $|C'| \leq |\mathcal{K}(G')| = |\mathcal{K}(G)| - 1$ . First, assume that  $y \notin C'$ . Then by the property of domination of  $C'$ , there exists a vertex  $z \in V(G')$  such that  $z \in N_{G'}(y) \cap C'$ . We claim that  $C = C' \cup \{x\}$  is an ID-code of  $G$ . First of all, that  $C$  is a dominating set of  $G$  is clear from the fact that  $C'$  is a dominating set of  $G'$ . To prove that  $C$  is a closed-separating set of  $G$ , we see that  $x$  is closed-separated in  $G$  from all vertices in  $V(G') \setminus \{y\}$  by itself and is closed-separated in  $G$  from  $y$  by the vertex  $z \in C'$ . Moreover, all other pairs of distinct vertices that are closed-separated by  $C'$  are also closed-separated by  $C$ . Thus,  $C$ , indeed, is an ID-code of  $G$ . This implies that  $\gamma^{\text{ID}}(G) \leq |C| \leq |\mathcal{K}(G)|$ , contrary to our assumption.

We therefore assume that  $y \in C'$ . If again, there exists a vertex  $z \in N_{G'}(y) \cap C'$ , then by the same reasoning as above,  $C = C' \cup \{x\}$  is an ID-code of  $G$ . Otherwise, we have  $N[y] \cap C' = \{y\}$ . Now, since  $G$  is connected, we have  $\deg_G(y) > 1$  and therefore, there exists a vertex  $w \in N_G(y) \setminus \{x\}$ . Then  $C = C' \cup \{w\}$  is an ID-code of  $G$ . This is because, first of all,  $C$  still closed-separates every pair of distinct vertices in  $V(G')$ . The vertex  $x$  is closed-separated from  $y$  by the vertex  $w \in C$ ; and from  $w$  by  $w$  itself. Moreover, for any vertex  $v$  in  $V(G') \setminus \{y, w\}$ ,  $v$  is closed-separated from  $y$  in  $G'$  by some vertex  $u_v \in N_{G'}[v] \cap C'$ . Then  $x$  is closed-separated from all such  $v$  in  $V(G') \setminus \{y, w\}$  by the vertices  $u_v \in C'$ . Moreover,  $C$  is clearly also a dominating set of  $G$ . Hence, this leads to the same contradiction as before. ◀

► **Case 6:  $G'$  has closed-twins.**

Assume that vertices  $u, v \in V(G')$  are a pair of closed-twins of  $G'$ . Since  $u$  and  $v$  were not closed-twins in  $G$ , it means that  $x$  is adjacent to, say,  $u$ , without loss of generality. This implies that  $u = y$ . Note that  $v$  is then unique with respect to being a closed-twin with  $y$  in  $G'$ . This is because, if  $u$  and some vertex  $v' (\neq v) \in V(G')$  were also closed-twins in  $G'$ , then it would mean that  $v$  and  $v'$  were closed-twins in  $G$ , contrary to our assumption. Now, let  $G'' = G' - v$ . We claim the following.

■ **Claim 1.**  $G''$  is closed-twin-free.

*Proof of claim.* Toward a contradiction, if vertices  $z, w \in V(G'')$  were a pair of closed-twins in  $G''$ , it would then mean that the vertex  $z \in N_{G'}(v)$ , without loss of generality. This would, in turn, imply that  $z \in N_{G'}(y)$  (since the vertices  $y$  and  $v$  are closed-twins in  $G'$ ). Or, in other words,  $y \in N_{G''}(z)$ . Now, since  $z$  and  $w$  are closed-twins in  $G''$ , we have  $y \in N_{G''}(w)$ , i.e.  $w \in N_{G'}(y)$ . Again, by virtue of  $y$  and  $v$  being closed-twins in  $G'$ , we have  $w \in N_{G'}(v)$ . This implies that  $z$  and  $w$  are closed-twins in  $G$  which is a contradiction to our assumption. ■

We also note here that the vertices  $y$  and  $v$  must be from the same block, as the two are adjacent on account of being closed-twins in  $G'$ . Thus,  $G''$  is a connected closed-twin-free block graph. Therefore, by the minimality of the order of  $G$ , there is an ID-code  $C''$  of  $G''$  such that  $|C''| \leq |\mathcal{K}(G'')| < |\mathcal{K}(G)|$ . If  $y \notin C''$ , then we claim that  $C = C'' \cup \{x\}$  is an identifying code of  $G$ . This is true because, firstly,  $C$  is a dominating set of  $G$  (note that, by the property of domination of  $C''$  in  $G''$ , there exists a vertex  $z \in N_{G''}(y) \cap C''$ ; and since  $y$  and  $v$  are closed-twins in  $G'$ , we have  $z \in N_G(v) \cap C$ ). Moreover,  $x$  is closed-separated in  $G$  from every other vertex in  $V(G) \setminus \{y\}$  by  $x$  itself; and the vertices  $x$  and  $y$  are closed-separated in  $G$  by some vertex in  $N_{G''}(y) \cap C''$  that dominates the vertex  $y$ . The vertex  $y$  is closed-separated from all the vertices in  $V(G) \setminus \{y, x\}$  by  $x$  and; since  $y$  and  $v$  have the same closed neighborhood in  $G'$ ,  $v$  is closed-separated in  $G$  from all vertices in  $V(G'') \setminus \{y\}$  because  $y$  is by  $C''$ . Finally, any two distinct vertices closed-separated by  $C''$  still remain so by  $C$ . Thus,  $C$ , indeed, is an ID-code of  $G$ . This implies that  $\gamma^{\text{ID}}(G) \leq |C| \leq |\mathcal{K}(G)|$ ; again a contradiction.

Let us, therefore, assume that  $y \in C''$ . This time, we claim that  $C = (C'' \setminus \{y\}) \cup \{x, v\}$  is an ID-code of  $G$ . That  $C$  is a dominating set of  $G$  is clear. So, as for the closed-separating property of  $C$  is concerned, as before,  $x$  is closed-separated in  $G$  from every other vertex in  $V(G) \setminus \{y\}$  by  $x$  itself; vertices  $x$  and  $y$  are closed-separated in  $G$  by  $v$ ; the vertices  $y$  and  $v$  are closed-separated in  $G$  by  $x$  and the vertices  $v$  and  $x$  are closed-separated in  $G$  by  $v$ . Since  $y$  and  $v$  have the same closed neighborhood in  $G'$  and since  $y$  is closed-separated in  $G''$  by  $C''$  from every other vertex in  $V(G'')$ , both  $v$  and  $y$  are also each closed-separated in  $G$  from every vertex in  $V(G'') \setminus \{v, y\}$ .

Finally, any two distinct vertices of  $G''$  are closed-separated by  $C''$  still remain so by  $C$ . This proves that  $C$  is an ID-code of  $G$  and hence, again, we are led to the contradiction that  $\gamma^{\text{ID}}(G) \leq |C| \leq |\mathcal{K}(G)|$ . ◀

This proves the theorem. □

Regarding the tightness of the bound in Theorem 5.1, note that, besides for stars [110], the upper bound is attained by the ID-numbers of thin headless spiders [5] which, therefore, serve as examples of cases where the bound in Theorem 5.1 is tight.

## 5.1.2 Lower bounds on ID-numbers of block graphs

We now turn to studying lower bounds on the ID-numbers of block graphs. We find the lower bounds of ID-numbers both in terms of the number of vertices and in terms of the number of blocks of a block graph. For the rest of this section, given a block graph  $G$ , by  $\mathcal{K}_{\text{leaf}}(G)$  we denote the set of all leaf blocks of  $G$  with at least one edge in the block. Moreover, by the symbol  $n_i(G)$ , we denote the number of vertices of degree  $i$  in the graph  $G$ .

### Lower bound in terms of the order of the graph

In terms of the order of the graph, we prove the following lower bound on the ID-number of an ID-admissible block graph.

**Theorem 5.2.** *Let  $G$  be a connected closed-twin-free block graph. Then we have*

$$\gamma^{\text{ID}}(G) \geq \frac{|V(G)|}{3} + 1.$$

To prove the theorem, we first go through some definitions and establish some lemmas. To that end, we also recall some definitions and the statements of some lemmas and corollaries that we had already encountered in Section 4.1.2 and which will again be used in this section.

**Lemma 4.4.** *Let  $G$  be a connected block graph with at least one edge. Then we have*

$$|\mathcal{K}(G)| \leq |V(G)| - 1 - |\mathcal{K}_{leaf}(G)| + n_1(G).$$

**Corollary 4.3.** *Let  $G$  be a block graph with  $k$  components. Then we have*

$$|\mathcal{K}(G)| - n_0(G) \leq |V(G)| - k.$$

**Definition 4.1.** For a given code  $C$  (not necessarily an ID-code) of a connected block graph  $G$ , let us assume that the sets  $C_1, C_2, \dots, C_k$  partition the code  $C$  such that the induced subgraphs  $G[C_1], G[C_2], \dots, G[C_k]$  of  $G$  are the  $k$  components of the subgraph  $G[C]$  of  $G$  induced by  $C$ . Note that each  $C_i$  is a block graph (since every induced subgraph of a block graph is also a block graph). Then, the vertex set  $V(G)$  is partitioned into the four following parts.

- (1)  $V_1 = C$ ,
- (2)  $V_2 = \{v \in V(G) \setminus V_1 : |N_G(v) \cap C| = 1\}$ ,
- (3)  $V_3 = \{v \in V(G) \setminus V_1 : \text{there exist distinct } i, j \leq k \text{ such that } N_G(v) \cap C_i \neq \emptyset \text{ and } N_G(v) \cap C_j \neq \emptyset\}$ ,
- (4)  $V_4 = V(G) \setminus (V_1 \cup V_2 \cup V_3)$ . Note that, for all  $v \in V_4$ , we have  $N_G(v) \cap C \subset C_i$  for some  $i$  and that  $|N_G(v) \cap C_i| \geq 2$ .

**Lemma 4.7.** *Let  $G$  be a connected block graph,  $C$  be a code (not necessarily an ID-code) of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Then, we have  $|V_3| \leq k - 1$ .*

We now prove a series of lemmas establishing upper bounds on the orders of the vertex subsets  $V_2$  and  $V_4$  of a connected block graph  $G$ .

**Lemma 5.1.** *Let  $G$  be a connected closed-twin-free block graph and  $C$  be an ID-code of  $G$ . Then, we have  $|V_2| \leq |C| - n_0(G[C])$ .*

*Proof.* By definition of  $V_2$ , each vertex  $v \in V_2$  has a unique neighbor  $u$  in  $C$ , that is,  $N_G(v) \cap C = \{u\}$ . Hence, there can be at most  $|C|$  vertices in  $V_2$ . Moreover, since  $C$  is an ID-code,  $u$  cannot be isolated in  $G[C]$  (or else,  $u$  and  $v$  will not be closed-separated in  $G$  by  $C$ ). Thus, there are at most  $|C| - n_0(G[C])$  vertices in  $V_2$  and this proves the result.  $\square$

**Lemma 5.2.** *Let  $G$  be a connected closed-twin-free block graph,  $C$  be an ID-code of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Then, we have  $|V_4| \leq |\mathcal{K}(G[C])| - |\mathcal{K}_{leaf}(G[C])| \leq |C| - 3k$ .*

*Proof.* Let  $v$  be any vertex in  $V_4$  and let  $G[C_i]$  be the component of  $G[C]$  such that  $N_G(v) \cap C \subseteq C_i$ . Moreover,  $|N_G(v) \cap C_i| \geq 2$ . Then, notice that  $N_G(v) \cap C$  must be a subset of exactly one block of  $G[C_i]$ , or else,  $G[C_i]$  would be disconnected, as  $v \notin C$ . This implies that  $|V_4| \leq |\mathcal{K}(G[C])| - n_0(G[C]) \leq |C| - k$ , by Corollary 4.3.

Now, let  $G[C_i]$  be a component of  $G[C]$  such that at least one vertex of  $V_4$  is adjacent to some vertices in  $C_i$ . In particular,  $|C_i| \geq 2$  and since  $C$  is an ID-code,  $G[C_i]$  is closed-twin-free. We first show the following.

■ **Claim 1.** No vertex in  $V_4$  is adjacent to the vertices of the leaf blocks of  $G[C_i]$ .



*Proof of claim.* Suppose that  $L$  is a leaf block of  $G[C_i]$ . Then we must have  $L \cong K_2$ , or else, at least two vertices in  $V(L)$  are not closed-separated in  $G$  by  $C$ . So, assume that  $V(L) = \{x, y\}$ . Then at least one of  $x$  and  $y$  must be a non-articulation vertex of  $G[C_i]$ . Without loss of generality, suppose that  $y$  is a non-articulation vertex of  $G[C_i]$ . If there exists a vertex  $v$  of  $V_4$  such that  $N_G(v) \cap C = V(L)$ , then  $v$  and  $y$  would not be closed-separated in  $G$  by  $C$  which is a contradiction. Hence, no element of  $V_4$  is adjacent to the vertices of the leaf blocks of  $G[C_i]$ . ■

This implies that the number of vertices of  $V_4$  that can be adjacent to the vertices of  $G[C_i]$  are at most  $|\mathcal{K}(G[C_i])| - |\mathcal{K}_{leaf}(G[C_i])|$ . Now, we must have the following.

■ **Claim 2.**  $|\mathcal{K}_{leaf}(G[C_i])| \geq 2$

*Proof of claim.* If, on the contrary,  $|\mathcal{K}_{leaf}(G[C_i])| = 1$ , then  $|C_i| = 1$ ; or else, all pairs of vertices of  $G[C_i]$  are not closed-separated in  $G$  by  $C$ . This contradicts the fact that  $|C_i| \geq 2$ . ■

Therefore, by the above two claims and the fact that any vertex of  $V_4$  having its neighbors in a component  $G[C_i]$  of  $G[C]$  is adjacent to the vertices of exactly one block of  $G[C_i]$ , the number of vertices of  $V_4$  adjacent to the vertices of  $G[C_i]$  is at most  $|\mathcal{K}(G[C_i])| - |\mathcal{K}_{leaf}(G[C_i])| \leq |\mathcal{K}(G[C_i])| - 2 \leq |C_i| - 3$  (the last inequality is by the fact that  $G[C_i]$  is connected, that is,  $k = 1$  and  $n_0(C_i) = 0$  in Corollary 4.3). Hence,

$$|V_4| \leq \sum_{1 \leq i \leq k} (|\mathcal{K}(G[C_i])| - |\mathcal{K}_{leaf}(G[C_i])|) \leq \sum_{1 \leq i \leq k} (|C_i| - 3) = |C| - 3k.$$

□

This brings us to the proof of Theorem 5.2.

**Theorem 5.2.** *Let  $G$  be a connected closed-twin-free block graph. Then we have*

$$\gamma^{\text{ID}}(G) \geq \frac{|V(G)|}{3} + 1.$$

*Proof.* Let  $|V(G)| = n$ . Assume  $C$  to be an ID-code of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Recalling from Definition 4.1 the sets  $V_1, V_2, V_3, V_4$  that partition  $V(G)$ , we prove the theorem using the relation  $|V(G)| = |C| + |V_2| + |V_3| + |V_4|$  and the upper bounds for  $|V_2|$ ,  $|V_3|$  and  $|V_4|$  in Lemmas 5.1, 4.7 and 5.2, respectively. Therefore, we have

$$\begin{aligned} n &= |C| + |V_2| + |V_3| + |V_4| \\ &\leq |C| + |C| - n_0(G[C]) + k - 1 + |C| - 3k \\ &\leq 3|C| - 2k - 1 \leq 3|C| - 3, \quad \text{using } k \geq 1. \end{aligned}$$

Hence, the result holds. □

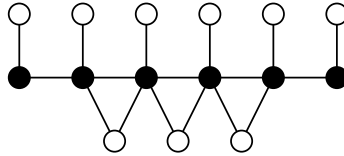


Figure 5.1: Extremal cases where the lower bounds in Theorem 5.2 are attained. The black vertices form a minimum ID-code.

We now look at examples of connected block graphs that are extremal with respect to Theorem 5.2. To that end, we prove the following proposition.

**Proposition 5.1.** *For  $k \geq 3$ , consider a path on vertices  $u_1, u_2, \dots, u_\ell$ . For all  $1 \leq i \leq \ell$ , attach a vertex  $v_i$  by the edge  $v_i u_i$  and, for each pair  $u_i, u_{i+1}$  for  $\ell \geq 4$  and  $2 \leq i \leq \ell - 2$ , attach a vertex  $w_i$  by edges  $w_i u_i$  and  $w_i u_{i+1}$ . See Figure 5.1 with  $\ell = 6$ . Then, such graphs are the only ones whose ID-numbers attain the lower bound in Theorem 5.2.*

*Proof.* For extremal graphs whose ID-numbers attain the lower bound in Theorem 5.2, we have equalities in the equations in the proof of Theorem 5.2 when  $C$  is an ID-code. We therefore have in this case  $k = 1$  (that is,  $G[C]$  is connected) which implies that  $|V_2| = |C|$ ,  $|V_3| = 0$  and  $|V_4| = |C| - 3$ . Tracing back the equality for  $|V_4|$ , it stems from equality in the statement of Lemma 5.2, that is,

$$|V_4| = |\mathcal{K}(G[C])| - |\mathcal{K}_{leaf}(G[C])| = |C| - 3k.$$

This further implies  $|\mathcal{K}_{leaf}(G[C])| = 2$  in Claim B of the proof of Lemma 5.2. Recall that  $k = 1$ . Hence, putting these values in the preceding equation, we have  $|V_4| = |\mathcal{K}(G[C])| - 2 = |C| - 3$  or  $|\mathcal{K}(G[C])| = |C| - 1$ . This implies that  $G[C]$  must be a tree and more particularly, a path, since  $|\mathcal{K}_{leaf}(G[C])| = 2$ . Moreover, with the values of  $|V_2| = |C|$  and  $|V_4| = |C| - 3$  coupled with the fact that no vertex of  $V_4$  can have neighbors in leaf blocks of  $C$  (Claim 1 in proof of Lemma 5.2), any extremal graph with respect to the lower bound for ID-numbers in Theorem 5.2 must be as described in the statement of the proposition.  $\square$

### Lower bound in terms of the number of blocks

With respect to the number of block, we prove the following lower bound on the ID-number of an ID-admissible block graph.

**Theorem 5.3.** *Let  $G$  be a connected closed-twin-free block graph and let  $\mathcal{K}(G)$  be the set of all blocks of  $G$ . Then we have*

$$\gamma^{\text{ID}}(G) \geq \frac{3(|\mathcal{K}(G)| + 2)}{7}.$$

To prove the theorem, we again recall some definitions and lemmas (without proof) established in Section 4.1.2.

**Definition 4.3.** Let  $C$  be a code (not necessarily an ID-code) of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . First, we define  $\mathcal{I}_G(C) = \{K \in \mathcal{K}(G) : V(K) \subset V(L) \text{ for some } L \in \mathcal{K}(G[C])\}$ . Moreover, for each  $1 \leq i \leq k$ , let  $\mathcal{I}_G(C_i) = \{K \in \mathcal{K}(G) : V(K) \subset V(L) \text{ for some } L \in \mathcal{K}(G[C_i])\}$ . Next, we define the following types of blocks of  $G$ .

- (1) Let  $\mathcal{K}_C(G) = \{K \in \mathcal{I}_G(C) : V(K) \subset C\}$ , i.e. all blocks of  $G$  which are also blocks of the subgraph  $G[C]$  (also a block graph) of  $G$ .
- (2) Let  $\mathcal{K}_{\overline{C}}(G) = \mathcal{K}(G) \setminus \mathcal{I}_G(C)$ . In other words, the set  $\mathcal{K}_{\overline{C}}(G)$  includes all blocks of  $G$  which do not contain any vertices of the code  $C$ .
- (3) For  $i = 2, 3, 4$ , let  $\mathcal{K}_i(G) = \{K \in \mathcal{I}_G(C) : V(K) \cap V_i \neq \emptyset\}$  (recall the sets  $V_1, V_2, V_3, V_4$  from Definition 4.1).

**Lemma 4.9.**  $|\mathcal{K}_2(G)| \leq |V_2|$ .

**Lemma 4.10.**  $|\mathcal{K}_3(G)| \leq 2(k - l)$ .

**Lemma 4.11.**  $|\mathcal{K}_{\overline{C}}(G)| \leq l - 1$ .

**Lemma 4.12.**  $|\mathcal{K}_C(G) \cup \mathcal{K}_4(G)| \leq |\mathcal{K}(G[C])| - n_0(G[C])$ .

This leads us to the proof of Theorem 5.3.

**Theorem 5.3.** *Let  $G$  be a connected closed-twin-free block graph and  $\mathcal{K}(G)$  be the set of all blocks of  $G$ . Then we have*

$$\gamma^{\text{ID}}(G) \geq \frac{3(|\mathcal{K}(G)| + 2)}{7}.$$

*Proof.* Let  $C$  be an ID-code of the block graph  $G$ . Then, we have

$$\mathcal{K}(G) = (\mathcal{K}_C(G) \cup \mathcal{K}_4(G)) \cup \mathcal{K}_{\overline{C}}(G) \cup \mathcal{K}_2(G) \cup \mathcal{K}_3(G).$$

Therefore, using Lemma 4.9, 4.10, 4.11 and 4.12, we have

$$\begin{aligned} |\mathcal{K}(G)| &\leq |\mathcal{K}_C(G) \cup \mathcal{K}_4(G)| + |\mathcal{K}_{\overline{C}}(G)| + |\mathcal{K}_2(G)| + |\mathcal{K}_3(G)| \\ &\leq |\mathcal{K}(G[C])| - n_0(G[C]) + l - 1 + |V_2| + 2(k - l) \\ &\leq |C| - k + l - 1 + |C| - n_0(G[C]) + 2(k - l) \quad [\text{using Corollary 4.3 and Lemma 5.1}] \\ &= 2|C| + k - l - n_0(G[C]) - 1 \\ &\leq 2|C| + k - n_0(G[C]) - 2. \end{aligned}$$

This proves the theorem.  $\square$

Note that, for any tree  $G$  (which are particular block graphs with each block being of order 2), we have  $|\mathcal{K}(G)| = |E(G)| = |V(G)| - 1$ . Therefore, Theorem 5.3 provides the same lower bound  $\left(\frac{3(|V(G)|+1)}{7}\right)$  for ID-numbers as was given in [25]. As far as tightness of Theorem 5.2 is concerned, for  $k = 3$  in Proposition 5.1, the graph  $G$  is a 1-corona of a  $P_3$  (with  $|\mathcal{K}(G)| = 5$ ) and is an extremal example whose ID-number ( $= 3$ ) attains the bound in Theorem 5.3. Apart from this example, there are infinite families of trees reaching the bound in Theorem 5.3 ([25]).

## 5.2 Graphs of a given maximum degree

Bounds on domination numbers for graphs with restrictions on their degree parameters are a natural and important line of research. In 1996, Reed [181] proved the influential result that if  $G$  is a connected cubic graph of order  $n$ , then  $\gamma(G) \leq \frac{3}{8}n$ . In 2009, this bound was improved to  $\gamma(G) \leq \frac{5}{14}n$ , if we exclude the two non-planar cubic graphs of order 8 [154]. A study of independent domination in graphs with bounded maximum degree has received considerable attention in the literature. We refer to the breakthrough paper [63], as well as the papers in [62, 80, 114]. Another classical result is that the total domination number of a cubic graph is at most one-half its order [3], and the remarkable result that the total domination number of a 4-regular graph is at most three-sevenths its order was proved by an interplay with the notion of transversals in hypergraphs [196]. A detailed discussion on upper bounds on domination parameters in graphs in terms of their order and minimum and maximum degree, as well as bounds with specific structural restrictions imposed, can be found in [125, Chapters 6, 7, 10].

With respect to the identification number, it was observed in [90] that when the maximum degree  $\Delta$  of the graph  $G$  is small enough with respect to the order  $n$  of the graph, the  $(n - 1)$ -upper bound can be significantly improved (for connected graphs) to  $n - \frac{n}{\Theta(\Delta^5)}$ . The latter was thereafter subsequently reduced to  $n - \frac{n}{\Theta(\Delta^3)}$  in [104]. This raises the question of what is the largest possible identification number of a connected identifiable graph of order  $n$  and maximum degree  $\Delta$ . Towards this problem, Conjecture 5.2 was posed. We recall the said conjecture below.

**Conjecture 5.2.** [99] *There exists a constant  $c$  such that for every connected identifiable graph of order  $n \geq 2$  and of maximum degree  $\Delta \geq 2$ ,*

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta - 1}{\Delta}\right)n + c.$$

Note that for  $\Delta \leq 1$ , the only connected identifiable graph is the one-vertex graph. From the known results in the literature, the above conjecture holds for  $\Delta = 2$  (that is, for paths and cycles) with  $c = 3/2$  (see Theorems 5.4 and 5.5). Hence, for the rest of the chapter, we assume that  $\Delta \geq 3$ .

If true, Conjecture 5.2 would be tight: For  $\Delta = 2$ , the conjecture is tight for both paths and cycles with  $c \leq 3/2$  (see corollary 5.1). For  $\Delta = 3$ , the conjecture is tight, for example, for trees presented

in Figure 5.2 and for a path whose every vertex we join to a 2-path by a single edge. For any  $\Delta > 3$ , the complete bipartite graph  $K_{\Delta, \Delta}$  satisfies  $\text{ID}(K_{\Delta, \Delta}) = 2\Delta - 2 = \left(\frac{\Delta-1}{\Delta}\right)n$  and hence, gives tight examples with  $c = 0$ . Furthermore, for any  $\Delta > 3$  and an unbounded value of  $n$ , there are trees with identification number  $\text{ID}(T) > \frac{(\Delta-1)n}{\Delta} - \frac{n}{\Delta^2}$  (see Proposition 5.4 and Figure 5.7). Moreover, for any value of  $\Delta \geq 3$ , there are arbitrarily large graphs of order  $n$  and maximum degree  $\Delta$  with identification number  $\left(\frac{\Delta-1}{\Delta}\right)n$ , see [104, 117]. A bound of the form  $n - \frac{n}{103(\Delta+1)^3}$  [104] proved using probabilistic arguments is the best known general result towards Conjecture 5.2 (for the sake of comparison, the conjectured bound can be rewritten as  $n - \frac{n}{\Delta} + c$ ). It is reduced to  $n - \frac{n}{103\Delta}$  for graphs with no forced vertices and to  $n - \frac{n}{f(k)\Delta}$  for graphs of clique number  $k$  [104]. For triangle-free graphs, this was improved to  $n - \frac{n}{\Delta + o(\Delta)}$  in [99], and to smaller bounds for subfamilies of triangle-free graphs, such as  $n - \frac{n}{\Delta+9}$  for bipartite graphs and  $n - \frac{n}{3\Delta/(\ln \Delta - 1)}$  for triangle-free graphs without (open) twins. The latter result implies that Conjecture 5.2 holds for triangle-free graphs without any open twins, whenever  $\Delta \geq 55$  (because then,  $3\Delta/(\ln \Delta - 1) \leq \Delta$ ). Conjecture 5.2 is also known to hold for line graphs of graphs of average degree at least 5 [91] as well as graphs which have girth at least 5, minimum degree at least 2 and maximum degree at least 4 [18]. Moreover, it holds for bipartite graphs without (open) twins by [100]. Furthermore, the conjecture holds in many cases for some graph products such as Cartesian and direct products [115, 145, 178]. See also the book chapter [60], where Conjecture 5.2 is presented.

All results in this section also appear in [41] and [42].

## 5.2.1 Paths and cycles

In this section, we recall results on all connected graphs with  $\Delta = 2$ , that is, on paths and cycles. The ID-number of all identifiable paths (that is, of all paths except  $P_2$ ) was determined by Bertrand et al. [24]. Moreover, using an upper bound from [24] on even cycles of order at least 6, Gravier et al. [111] provided the exact values of the identification numbers of all identifiable cycles (that is, cycles of length at least 4). We summarize these results in the following theorems in comparison to the upper bound of the form  $\left(\frac{\Delta-1}{\Delta}\right)n + c$  as in Conjecture 5.2.

**Theorem 5.4** ([24]). *If  $P_n$  is a path on  $n$  vertices, then we have*

$$\text{ID}(P_n) = \begin{cases} \frac{n}{2} + \frac{1}{2}, & \text{if } n \geq 1 \text{ is odd,} \\ \frac{n}{2} + 1, & \text{if } n \geq 4 \text{ is even} \end{cases} \leq \left(\frac{\Delta-1}{\Delta}\right)n + 1.$$

**Theorem 5.5** ([111]). *If  $C_n$  is a cycle on  $n$  vertices, then we have*

$$\text{ID}(C_n) = \begin{cases} 3, & \text{if } n = 4, 5, \\ \frac{n}{2}, & \text{if } n \geq 6 \text{ is even,} \\ \frac{n}{2} + \frac{3}{2}, & \text{if } n \geq 7 \text{ is odd} \end{cases} \leq \left(\frac{\Delta-1}{\Delta}\right)n + \frac{3}{2}.$$

Using Theorems 5.4 and 5.5, therefore, one has the following corollary.

**Corollary 5.1.** *The following hold.*

- (a) *If  $G$  is a path, then  $\gamma^{\text{ID}}(P_n) = \lfloor \frac{n}{2} \rfloor + 1$ .*
- (b) *If  $G = C_4$  or  $G = C_5$ , then  $\gamma^{\text{ID}}(G) = \lfloor \frac{n}{2} \rfloor + 1$ .*
- (c) *If  $n \geq 6$  is even, then  $\gamma^{\text{ID}}(C_n) = \frac{n}{2}$ .*
- (d) *If  $n \geq 7$  is odd, then  $\gamma^{\text{ID}}(C_n) = \frac{n}{2} + \frac{3}{2}$ .*
- (e) *If  $n = 4$ , then  $\gamma^{\text{ID}}(P_n) = \frac{3}{4}n$ , and if  $n \geq 3$  and  $n \neq 4$ , then  $\gamma^{\text{ID}}(P_n) \leq \frac{2}{3}n$ .*
- (f) *If  $n \in \{4, 7\}$ , then  $\frac{2}{3}n < \gamma^{\text{ID}}(C_n) \leq \frac{3}{4}n$ , and if  $n \geq 3$  and  $n \notin \{4, 7\}$ , then  $\gamma^{\text{ID}}(C_n) \leq \frac{2}{3}n$ .*

We shall need the following elementary property of odd cycles with one edge added.

**Observation 5.1** (Proposition 4.1 of [195]). *If  $n = 2k + 1 \geq 7$  is odd,  $G = C_n$  and if  $e \in E(\overline{G})$ , then  $\gamma^{\text{ID}}(G + e) \leq k + 1 \leq \frac{2}{3}n$ .*

## 5.2.2 Trees of given maximum degree

One of the challenges of proving Conjecture 5.2 for trees is that we need to allow open twins of degree 1, which are present in many trees (note that for any set of mutual open twins, one needs all of them but one in any identifying code). We will also see that almost all extremal trees for Conjecture 5.2 (those requiring  $c > 0$ ) have twins. Moreover, it is known that when a tree  $T$  of order at least 3 (except the path  $P_4$ ) has no open twins, it satisfies  $\gamma^{\text{ID}}(T) \leq \frac{2}{3}n$  [100] (this even holds for bipartite graphs). This implies the conjecture for  $\Delta \geq 3$  for twin-free bipartite graphs, and clearly shows that the presence of open twins is the main difficulty in proving Conjecture 5.2 for trees.

As pointed out before, we focus on the maximum degree  $\Delta \geq 3$ . In our current work, we prove Conjecture 5.2 (with  $c = \frac{1}{\Delta} \leq \frac{1}{3}$ ) for all trees of maximum degree at least 3 (by Theorem 5.4, the conjecture already holds with  $c = 1$  for all identifiable trees of maximum degree 2, that is, for paths).

The main challenge for proving the conjecture, is the constant  $c$  that could, in principle, be arbitrary. Thus, in order to prove it for trees, a large part of our proof is dedicated to analyzing those extremal trees that require  $c > 0$ . In fact, for each  $\Delta \geq 3$ , we characterize the trees with maximum degree  $\Delta$  for which  $c > 0$ . The number of these trees is largest for  $\Delta = 3$ , and in fact, this case is the hardest part of our proof. The characterization is given by the collection  $\mathcal{T}_\Delta$  for  $\Delta \geq 3$ , where, for  $\Delta = 3$ ,  $\mathcal{T}_\Delta$  is the set of 12 trees of maximum degree 3 and diameter at most 6 depicted in Figure 5.2; and  $\mathcal{T}_\Delta = \{K_{1,\Delta}\}$  for  $\Delta \geq 4$ .

Note that, for maximum degree at least 3, all trees are identifiable. Hence, throughout the rest of the paper, we tacitly assume all our trees to be identifiable. Our main results for trees are stated as follows.

**Theorem 5.6.** *Let  $G$  be a tree of order  $n$  and of maximum degree  $\Delta \geq 3$ . If  $G$  is isomorphic to a tree in  $\mathcal{T}_\Delta$ , then, we have*

$$\gamma^{\text{ID}}(G) = \left( \frac{\Delta - 1}{\Delta} \right) n + \frac{1}{\Delta}.$$

*On the other hand, if  $G$  is not isomorphic to any tree in the collection  $\mathcal{T}_\Delta$ , then we have*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n.$$

We also determine the exact value of the ID-number for the exceptional trees in  $\mathcal{T}_\Delta$ , as follows. We have listed every tree requiring a positive constant  $c$  for Conjecture 5.2 in Table 5.1. We also show that Theorem 5.6 is tight for many trees with  $c = 0$ , besides the list of exceptional trees mentioned above which require  $c > 0$ . One tight example of trees with  $c = 0$  is the 2-corona of a path [100]. We will see that other such tight examples exist. When  $\Delta = 3$ , there are infinitely many examples for such trees. Furthermore, we give in Proposition 5.4 an infinite family of trees for any  $\Delta \geq 4$  which have identification number quite close to the conjectured bound, namely  $\frac{\Delta-1+\frac{1}{\Delta-2}}{\Delta+\frac{2}{\Delta-2}}n > (\frac{\Delta-1}{\Delta})n - \frac{n}{\Delta^2}$ . In particular, as  $\Delta$  increases, our construction gets closer and closer to the conjectured bound.

Graph family	$\Delta$	$c$	Reference
$K_{1,\Delta}$	$\Delta \geq 3$	$1/\Delta$	Lemma 5.5
$\mathcal{T}_3$	3	$1/3$	Theorem 5.6
Even paths	2	1	[24] (Theorem 5.4)
Odd paths	2	$1/2$	[24] (Theorem 5.4)

Table 5.1: Trees requiring a positive constant  $c$  for Conjecture 5.2.

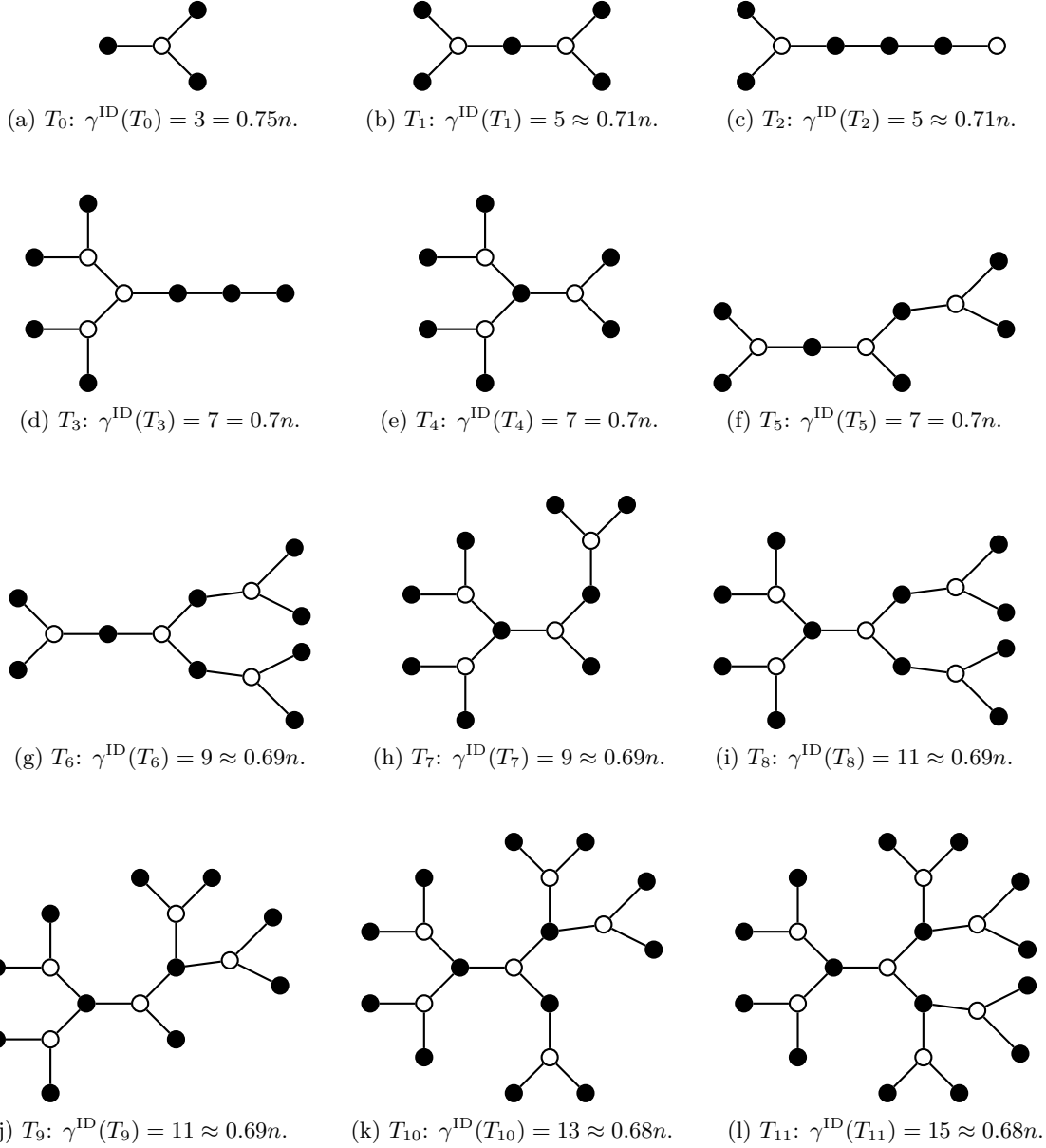


Figure 5.2: The family  $\mathcal{T}_3$  of trees of maximum degree 3 requiring  $c > 0$  in Conjecture 5.2. The set of black vertices in each figure constitutes an identifying code of the tree.

### 5.2.2.1 Preliminary results

The number of leaves and support vertices in a graph  $G$  are denoted by  $\ell(G)$  and  $s(G)$ , respectively. Naturally, any vertex of a graph  $G$  that is not a leaf of  $G$  is usually referred to as a *non-leaf* vertex of  $G$ . One can check that the following remark holds.

**Remark 5.3.** *Let  $G$  be an identifiable graph. A dominating set  $C$  of  $G$  is an identifying code of  $G$  if and only if  $C$  separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \leq 2$ .*

In light of the above Remark 5.3, on all occasions throughout this paper where we need to prove a dominating set  $C$  of an identifiable graph  $G$  to be an identifying code of  $G$ , we simply show that  $C$  separates all pairs of distinct vertices of  $G$  with distance at most 2.

The domination number of a path  $P_n$  on  $n$  vertices is given by the following closed formula, noting that starting from the second vertex of the path, we add every third vertex on the path into our dominating set, together with the last vertex when  $n$  is not divisible by 3.

**Observation 5.2** ([125]). *If  $P_n$  is a path on  $n$  vertices, then  $\gamma(P_n) = \lceil \frac{n}{3} \rceil \leq \frac{n+2}{3}$ .*

We cite here the following two lemmas from [100], that will be essential for the proof of our main result, Theorem 5.6. The first lemma was originally proved only for trees in [124].

**Lemma 5.3** ([100, Lemma 4]). *If  $G$  is a connected bipartite graph on  $n \geq 4$  vertices with  $s$  support vertices and not isomorphic to a path  $P_4$ , then  $\gamma^{\text{ID}}(G) \leq n - s$ .*

**Lemma 5.4** ([100, Theorem 6]). *If  $G$  is a connected bipartite graph on  $n \geq 3$  vertices and  $\ell$  leaves, and with no twins of degree 2 or greater, then  $\gamma^{\text{ID}}(G) \leq \frac{1}{2}(n + \ell)$ .*

Note that both Lemma 5.3 and Lemma 5.4 apply to trees, since trees are bipartite graphs in which the only possible twins are leaves sharing the same support vertex, and thus, they have degree 1.

### 5.2.2.2 An upper bound linking the ID-numbers and domination numbers of trees

In this section, we introduce a simple new upper bound for identifying codes in trees, which improves the upper bound  $\gamma^{\text{ID}}(T) \leq n - s$  from Lemma 5.3 (for trees), since in any graph of order at least 3, the domination number is at least the number of support vertices. Note that the bound of Lemma 5.3 was initially proved in [124] for trees only.

**Theorem 5.7.** *If  $T$  is a tree other than  $P_4$  of order  $n \geq 3$ , then  $\gamma^{\text{ID}}(T) \leq n - \gamma(T)$ .*

*Proof.* We prove the claim by induction on the order of the tree. But first, we show that the claim holds for paths. By Observation 5.2,  $\gamma(P_n) = \lceil \frac{n}{3} \rceil \leq \frac{n+2}{3}$ . Moreover, by Theorem 5.4, we have  $\gamma^{\text{ID}}(P_n) = \frac{1}{2}(n + 2)$  for even  $n$  and  $\gamma^{\text{ID}}(P_n) = \frac{1}{2}(n + 1)$  for odd  $n$ . Furthermore, we have  $\gamma^{\text{ID}}(P_n) + \gamma(P_n) \leq \frac{n+2}{2} + \frac{n+2}{3} = \frac{5n+10}{6} \leq n$ , when  $n \geq 10$  is even and  $\gamma^{\text{ID}}(P_n) + \gamma(P_n) \leq \frac{n+1}{2} + \frac{n+2}{3} = \frac{5n+7}{6} \leq n$ , when  $n \geq 7$  is odd. Hence, the claim follows for even-length paths on at least ten vertices and for odd-length paths on at least seven vertices. Furthermore, we have  $\gamma^{\text{ID}}(P_3) + \gamma(P_3) = 2 + 1 = 3$ ,  $\gamma^{\text{ID}}(P_5) + \gamma(P_5) = 3 + 2 = 5$ ,  $\gamma^{\text{ID}}(P_6) + \gamma(P_6) = 4 + 2 = 6$  and  $\gamma^{\text{ID}}(P_8) + \gamma(P_8) = 5 + 3 = 8$ . Hence, the claim follows for all paths.

For  $n = 3$ , the only tree is  $P_3$  which satisfies  $\gamma^{\text{ID}}(P_3) = 2$  and  $\gamma(P_3) = 1$ , and so the bound holds. For  $n = 4$ , if  $T$  is not  $P_4$ , then it is the star  $K_{1,3}$ , with  $\gamma^{\text{ID}}(K_{1,3}) = 3$  and  $\gamma(K_{1,3}) = 1$ . Hence, the claim is true for  $n \leq 4$ .

Let us next assume that the claimed bound holds for every tree  $T'$  other than  $P_1, P_2, P_4$  that has order at most  $n'$ . Suppose, to the contrary, that there are trees of order  $n' + 1$  for which the bound does not hold. Let  $T$  be a tree of order  $n = n' + 1 \geq 5$  such that  $\gamma^{\text{ID}}(T) > n - \gamma(T)$ . By the first part of the proof, the tree  $T$  is not a path.

If all vertices of  $T$  are either leaves or support vertices, then  $\gamma(T) \leq s(T)$  and  $n - \gamma(T) \geq n - s(T)$  and the bound of the statement holds by Lemma 5.3, a contradiction. Thus, there exists a non-support, non-leaf vertex in  $T$ . Let us root the tree at such a non-leaf, non-support vertex  $x$ , and let us consider the support vertex  $s$  which has the greatest distance to  $x$ . Notice that vertex  $s$  is adjacent to exactly one non-leaf vertex due to its maximal distance to  $x$ .

Let us first assume that  $s$  is adjacent to  $t \geq 2$  leaves  $l_1, l_2, \dots, l_t$ . Consider the tree  $T_s = T - \{s, l_1, l_2, \dots, l_t\}$ . Since  $T_s$  has  $n - t - 1$  vertices, by induction we have  $\gamma^{\text{ID}}(T_s) \leq n - t - 1 - \gamma(T_s)$  unless  $T_s = P_4$  or  $T_s$  contains at most two vertices. However, if  $T_s = P_4$ , then  $\gamma(T) = 2 = s(T)$  if vertex  $s$  is adjacent to a leaf of  $T_s$ , and if  $s$  is adjacent to a support vertex of  $T_s$ , then  $\gamma(T) = 3 = s(T)$ . Since  $\gamma^{\text{ID}}(T) \leq n - s(T)$  by Lemma 5.3, the claim holds in these cases. Furthermore, if  $T_s$  contains one or two vertices, then again  $\gamma(T) = s(T)$  and the claim follows. Therefore, we may assume by

induction that there exists an identifying code  $C_s$  in  $T_s$  which has cardinality at most  $n-t-1-\gamma(T_s)$ , with  $D_s$  a minimum-ordered dominating set in  $T_s$ . Furthermore, set  $C_s \cup \{l_1, \dots, l_t\}$  is an identifying code of  $T$ , and set  $D_s \cup \{s\}$  is a dominating set of  $T$ . Thus,  $\gamma^{\text{ID}}(T) \leq n-t-1-\gamma(T_s)+t \leq n-\gamma(T)$ , as claimed.

Let us assume next that support vertex  $s$  has degree 2, and denote by  $u$  the non-leaf adjacent to  $s$  (possibly,  $u = x$ ) and by  $l$  the leaf adjacent to  $s$ . If vertex  $u$  also has degree 2, then we consider the tree  $T_u = T - \{l, s, u\}$ . If  $T_u$  contains two or fewer vertices, then  $T$  is a path, a contradiction. Moreover, if  $T_u \cong P_4$ , then  $T$  is path if  $u$  is adjacent to a leaf of  $T_u$ , again a contradiction. Hence,  $T$  has three support vertices and  $\gamma^{\text{ID}}(T) \leq n-3$  (consider the identifying code formed by all non-leaf vertices of  $T$ ) while  $\gamma(T) = 3$ . Hence, by induction, we may assume that  $T_u$  has an optimal identifying code  $C_u$  with cardinality  $|C_u| = \gamma^{\text{ID}}(T_u) \leq n-3-\gamma(T_u)$  and we consider a minimum-ordered dominating set  $D_u$  of  $T_u$ . Notice that  $D_u \cup \{s\}$  is a dominating set in  $T$  and either  $C_u \cup \{s, u\}$  or  $C_u \cup \{l, u\}$  is an identifying code in  $T$ . Hence,  $\gamma^{\text{ID}}(T) \leq n-3-\gamma(T_u)+2 \leq n-\gamma(T)$  as claimed.

Therefore, we may assume from now on that  $\deg_T(u) \geq 3$ . Let us next consider the case where  $u$  is a support vertex with adjacent leaf  $l_u$ . Furthermore, let us consider the tree  $T_s = T - \{s, l\}$ . Notice that  $T_s$  contains at least two vertices, and if  $T_s$  contains only two vertices, then  $T$  is a path, a contradiction. Moreover, if  $T_s \cong P_4$ , then  $u$  is a support vertex of  $P_4$  and  $\gamma^{\text{ID}}(T) = \gamma(T) = 3 = n - \gamma(T)$  (consider the set of non-leaf vertices as an identifying code of  $T$ ). Hence, we may apply induction to  $T_s$  and thus, there exists an optimal identifying code  $C_s$  of  $T_s$  of cardinality  $|C_s| = \gamma^{\text{ID}}(T_s) \leq n-2-\gamma(T_s)$ . Also consider a minimum-ordered dominating set  $D_s$  in  $T_s$ . Assume first that  $u \in C_s$ . Now set  $C_s \cup \{l\}$  is an identifying code in  $T$  and  $D_s \cup \{s\}$  is a dominating set in  $T$ . Thus,  $\gamma^{\text{ID}}(T) \leq |C_s| + 1 \leq n-2-\gamma(T_s) + 1 \leq n-\gamma(T)$  and we are done. Moreover, if  $u \notin C_s$ , then  $l_u \in C_s$ . Furthermore, in this case set  $C = (C_s \cup \{s, u\}) \setminus \{l_u\}$  is an identifying code in  $T$ . Indeed, since  $C_s$  is an identifying code in  $T_s$  and  $N_{T_s}[l_u] \cap C_s = \{l_u\}$ , we have  $|N_{T_s}[u] \cap C_s| \geq 2$ . Hence, we have  $|N_T[u] \cap C| \geq 3$ ,  $N_T[l_u] \cap C = \{u\}$ ,  $N_T[s] \cap C = \{u, s\}$  and  $N_T[l] \cap C = \{s\}$ , confirming that  $C$  is an identifying code of  $T$ . Moreover, we have  $\gamma^{\text{ID}}(T) \leq n-\gamma(T)$  by the same arguments as in the case where  $u \in C_s$ .

Hence, we may assume that  $u$  is not a support vertex. Since we assumed that  $s$  is a support vertex with the greatest distance to  $x$ , there is at most one non-support vertex adjacent to  $u$  (on the path from  $u$  to  $x$ ). Moreover, if there exists a support vertex  $s' \in N(u)$  with  $\deg(s') \geq 3$ , then we could have considered  $s'$  instead of  $s$  in an earlier argument considering the case  $\deg(s) \geq 3$ . Thus, we may assume that every support neighbor of  $u$  has degree 2. Let us denote the support vertices adjacent to  $u$  by  $\{s_1, \dots, s_h\}$  (with  $s = s_1$ ) and the leaf adjacent to  $s_i$  by  $l_i$  for each  $1 \leq i \leq h$ . Observe that the tree  $SS_h = T[\{s_1, \dots, s_h, l_1, \dots, l_h, u\}]$  is a subdivided star. Moreover, it has domination number  $h$  and identification number  $h+1$ . Indeed, we may take the center and every support vertex as our identifying code  $C_h$ . Hence, the subdivided star  $SS_h$  satisfies the claimed upper bound:  $\gamma^{\text{ID}}(SS_h) = h+1 = 2h+1-\gamma(SS_h)$ . Observe that if  $T_{SS} = T - SS_h$  consists of two vertices, then  $T$  is a subdivided star  $SS_{h+1}$  and the claim follows. If  $T - SS_h$  contains only one vertex, then  $u$  was a support vertex, contradicting our assumption. Furthermore, if  $T_{SS} \cong P_4$ , then if  $u$  is adjacent to a support vertex of  $P_4$ , we may consider  $C_h$  together with the support vertices of  $P_4$  as our identifying code. If  $u$  is adjacent to a leaf of  $P_4$ , then we may consider  $C_h$  together with the leaf farthest from  $u$  and the only non-leaf vertex at distance 2 from  $u$ . In both cases, we have  $\gamma^{\text{ID}}(T) \leq h+3 = n-h-2$  since  $n = 2h+5$ , and  $\gamma(T) = h+2$ . Hence, the upper bound  $\gamma^{\text{ID}}(T) \leq n-\gamma(T)$  holds. Finally, there is the case where, by induction, the tree  $T_{SS}$  contains an identifying code  $C_{SS}$  of cardinality  $|C_{SS}| = \gamma^{\text{ID}}(T_{SS}) \leq n-(2h+1)-\gamma(T_{SS})$ . However, in this case, the set  $C_{SS} \cup C_h$  is an identifying code in  $T$  containing at most  $n-(2h+1)-\gamma(T_{SS})+(h+1) = n-(h+\gamma(T_{SS})) \leq n-\gamma(T)$  vertices. This completes the proof.  $\square$

Observe that for every tree  $T$  of order at least 3, we have  $\gamma(T) \geq s(T)$  since every leaf needs to be dominated by a vertex in the dominating set. Moreover, for example, in a path, we can have  $\gamma(P_n) > s(P_n)$ . Therefore, the upper bound of Theorem 5.7 is an improvement over the simple but useful upper bound  $n-s(T)$  from Lemma 5.3.



Moreover, the upper bound  $n - \gamma(T)$  is tight for example for every tree in Figure 5.2. This was the leading motivation for us to prove this bound. If we compare the upper bound of Theorem 5.6 to the upper bound  $n - \gamma(T)$  of Theorem 5.7, we observe that in some cases, they are equal (for example for bi-stars consisting of two equal-sized stars), but in some cases the domination bound gives a smaller value (for example for bi-stars consisting of two unequal stars). In some cases, the bound from Theorem 5.6 is even smaller (for example for a chain of equal-sized stars joined with a single edge between some leaves).

### 5.2.2.3 Characterizing extremal trees

As mentioned before, the main difficulty in proving Conjecture 5.2 lies in handling the constant  $c$ . In this section, we focus on those extremal trees that require  $c > 0$ , and by a careful analysis, we are able to fully characterize them.

Since the identification number of paths is well-understood, for the rest of the section, we only consider graphs of maximum degree  $\Delta \geq 3$ . Towards proving the first part of Theorem 5.6, we next look at the *star* graphs. For any  $\Delta \geq 3$ , the complete bipartite graph  $K_{1,\Delta}$  is called a  $\Delta$ -*star*, or simply a *star*. Noting that for any  $\Delta$ -star  $S$  the set of all its leaves constitutes a minimum identifying code of  $S$ , it can therefore be readily verified that the following lemma is true.

**Lemma 5.5.** *For a  $\Delta$ -star  $S$  with  $\Delta \geq 3$  of order  $n (= \Delta + 1)$ , we have*

$$\gamma^{\text{ID}}(S) = \left( \frac{\Delta - 1}{\Delta} \right) n + \frac{1}{\Delta}.$$

In particular, Lemma 5.5 shows that  $\Delta$ -stars satisfy the conjectured bound with  $c = \frac{1}{\Delta}$ . Hence, for all stars the constant  $c = \frac{1}{3}$  suffices in Conjecture 5.2. To fully establish the first part of Theorem 5.6 now, one needs to only show the veracity of the result for the rest of the trees in  $\mathcal{T}_3$ . To describe the trees in  $\mathcal{T}_3$  and other graphs later in a more unified manner, we start by defining a particular “join” of graphs with stars. Let  $G'$  be a graph and  $S$  be any star. Then, let  $G' \triangleright_v S$  denote the graph obtained by identifying a vertex  $v$  of  $G'$  with a leaf  $l$  of  $S$  (for example, if  $S$  and  $P$  are a 3-star and 4-path, respectively, each with a leaf  $v$ , then the graphs in Figures 5.2(a) and 5.2(b) are  $S \triangleright_v S$  and  $P \triangleright_v S$ , respectively). We call the  $G' \triangleright_v S$  the *graph  $G'$  appended with a star* and it is said to be obtained by *appending  $S$  (by its leaf  $l$ ) onto (the vertex  $u$  of)  $G'$* . In the case that the vertex  $v$  of  $G'$  is inconsequential to the context or is (up to isomorphism) immaterial to the graph  $G$ , we may simply drop the suffix  $v$  in the notation  $G' \triangleright_v S$  and denote it as  $G' \triangleright S$  (for example, if  $P$  is a 2-path and  $S$  is a star, then  $P \triangleright_v S$  is (up to isomorphism) the only graph irrespective of which vertex of the 2-path  $v$  is. Another example would be if  $S$  is a  $\Delta$ -star and we require  $S \triangleright_v S$  to be a graph of maximum degree  $\Delta$ . As a convention, we continue to call the vertices of  $G' \triangleright S$  by the same names as they were called in the graphs  $G'$  and  $S$ . In other words, the graph  $G' \triangleright S$  is said to *inherit* its vertices from  $G'$  and  $S$ . In particular, if  $G' \triangleright S$  is obtained by identifying the vertex  $v$  of  $G'$  and a leaf  $l$  of  $S$ , then both the names  $v$  and  $l$  (as and when convenient) also refer to the identified vertex in  $G' \triangleright S$ .

Let  $G_0$  be a fixed graph,  $p \geq 1$  be an integer and for each  $i \in [p]$ , let  $S_i$  be a  $\Delta_i$ -star for  $\Delta_i \geq 3$ . Now, we may carry out the process of inductively appending stars by defining  $G_i = G_{i-1} \triangleright_{v_{i-1}} S_i$  for all  $i \in [p]$ , where  $v_{i-1}$  is a vertex of  $G_{i-1}$ . Then the graph  $G_p$  is called the *graph  $G_0$  appended with  $p$  stars*. In the case that each  $S_i$  is isomorphic to a  $\Delta$ -star  $S$  for  $\Delta \geq 3$ , we call the graph  $G_p$  the *graph  $G_0$  appended with  $p$   $\Delta$ -stars*. In the particular case that  $G_0 = S_0$  is itself a  $\Delta_0$ -star for  $\Delta_0 \geq 3$ , we simply call  $G_p$  an *appended star*. Further, if  $\Delta = \Delta_0 = \Delta_1 = \dots = \Delta_p$ , then we call the graph  $G_p$  an *appended  $\Delta$ -star*.

We next furnish some general results for any identifiable graph appended with a star.

**Lemma 5.6.** *If  $G'$  is an identifiable graph and  $G = G' \triangleright_v S$ , where  $v$  is a vertex of  $G'$  and  $S$  is a  $\Delta$ -star for  $\Delta \geq 3$ , then  $G$  is also identifiable and*

$$\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(G') + \Delta - 1.$$

*Proof.* Graph  $G$  is clearly identifiable. Assume now that  $l_1, l_2, \dots, l_{\Delta-1}$  are the leaves that  $G$  inherits from  $S$ . Also, assume that  $u$  is the universal vertex of  $S$ , that is,  $u$  is the common support vertex of the leaves  $l_1, l_2, \dots, l_{\Delta-1}$  in  $G$ . Suppose now that  $C'$  is a minimum identifying code of  $G'$ . Then we claim that the set  $C = C' \cup \{l_1, l_2, \dots, l_{\Delta-1}\}$  is an identifying code of  $G$ . Clearly,  $C$  is a dominating set of  $G$ , since  $C'$  is a dominating set of  $G'$ . Next, to check that  $C$  is also a separating set of  $G$ , it is enough to check that each of the vertices  $u, l_1, l_2, \dots, l_{\Delta-1}$  is separated from all vertices at distance at most 2 in  $G$  by  $C$ . To start with, assume  $w$  to be a vertex of  $G$  other than  $u$ . Then, one can find at least one  $l_i$  to always be an separating  $C$ -codeword for the pair  $u, w$ . Moreover, each  $l_i$  is itself an separating  $C$ -codeword for the pair  $l_i, w$ , where  $l_i \neq w$ . This establishes the claim that  $C$  is an identifying code of  $G$ . Therefore,  $\gamma^{\text{ID}}(G) \leq |C| = |C'| + \Delta - 1 = \gamma^{\text{ID}}(G') + \Delta - 1$ . This proves the result.  $\square$

The next lemma shows that if  $G$  is the graph obtained by starting from an identifiable "base" graph  $G_0$  and iteratively appending stars thereon, then the graphs  $G$  and  $G_0$  share the same constant  $c$  in Conjecture 5.2.

**Lemma 5.7.** *Let  $c$  be a constant,  $G_0$  be an identifiable graph of order  $n_0$ , of maximum degree  $\Delta_0$  and be such that  $\gamma^{\text{ID}}(G_0) \leq \left(\frac{\Delta_0-1}{\Delta_0}\right)n_0 + c$ . For an integer  $p \geq 1$  and  $i \in [p]$ , let  $S_i$  be a  $\Delta_i$ -star for  $\Delta_i \geq 3$ . Also, for  $i \in [p]$ , let  $G_i = G_{i-1} \triangleright_{v_{i-1}} S_i$ , where  $v_{i-1}$  is a vertex of  $G_{i-1}$ , and  $G = G_p$  is a graph of order  $n$  and maximum degree  $\Delta$  obtained by appending  $p$  stars to  $G_0$ . Moreover, assume that  $\Delta_{\max} = \max\{\Delta_i : 0 \leq i \leq p\}$ . Then, we have*

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta_{\max}-1}{\Delta_{\max}}\right)n + c \leq \left(\frac{\Delta-1}{\Delta}\right)n + c.$$

*Proof.* Let us prove the claim by induction on  $q \in [0, p]$ . Case  $q = 0$  follows from the statement itself regarding the graph  $G_0$ . Let us assume that the induction hypothesis holds for  $q \in [0, p-1]$  and consider the case that  $q = p \geq 1$ . We have  $G = G_{p-1} \triangleright S_p$ , where we let  $G_{p-1}$  have order  $n'$  and where  $\Delta'_{\max} = \max\{\Delta_i : 1 \leq i \leq p-1\}$ . Finally, by the induction hypothesis, let  $C'$  be a minimum identifying code of  $G_{p-1}$  of cardinality at most  $\left(\frac{\Delta'_{\max}-1}{\Delta'_{\max}}\right)n' + c$ . By Lemma 5.6, there exists an identifying code  $C$  of  $G$  of cardinality  $|C'| + \Delta_p - 1$ . Therefore, we have

$$\begin{aligned} \gamma^{\text{ID}}(G) \leq |C| &= |C'| + \Delta_p - 1 \leq \left(\frac{\Delta'_{\max}-1}{\Delta'_{\max}}\right)n' + c + \left(\frac{\Delta_p-1}{\Delta_p}\right)\Delta_p \\ &\leq \left(\frac{\Delta_{\max}-1}{\Delta_{\max}}\right)(n - \Delta_p) + \left(\frac{\Delta_{\max}-1}{\Delta_{\max}}\right)\Delta_p + c \\ &= \left(\frac{\Delta_{\max}-1}{\Delta_{\max}}\right)n + c \\ &\leq \left(\frac{\Delta-1}{\Delta}\right)n + c. \end{aligned}$$

$\square$

**Corollary 5.2.** *For an integer  $p \geq 1$  and  $i \in [0, p]$ , let  $S_i$  be a  $\Delta_i$ -star for  $\Delta_i \geq 3$ . For  $i \in [p]$ , let  $G_i = G_{i-1} \triangleright_{v_{i-1}} S_i$ , where  $v_{i-1}$  is a vertex of  $G_{i-1}$ , and where  $G = G_p$  is an appended star of order  $n$  and of maximum degree  $\Delta$ . Moreover, assume that  $\Delta_{\max} = \max\{\Delta_i : 0 \leq i \leq p\}$ . Then, we have*

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta_{\max}-1}{\Delta_{\max}}\right)n + \frac{1}{\Delta_0} \leq \left(\frac{\Delta-1}{\Delta}\right)n + \frac{1}{3}.$$

*Proof.* The result follows from taking  $G_0 = S_0$  in Lemma 5.7 and  $c = \frac{1}{\Delta_0} \leq \frac{1}{3}$  from Lemma 5.5.  $\square$

It is worth mentioning here that Lemma 5.7 is central to our inductive proof arguments later, whereby, if a graph  $G$  is structurally isomorphic to a "smaller" identifiable graph  $G'$  appended with a star, then by using Lemma 5.7 for  $p = 1$  and an inductive hypothesis that the "smaller" graph

$G' = G - S$  satisfies the conjectured bound, one can show that so does the "bigger" graph  $G$ . We next look at the identification numbers of the trees isomorphic to  $P \triangleright_v S$ , where  $P$  is a path,  $v$  is a vertex of  $P$  and  $S$  is a star. In particular, we establish the identification numbers of the trees  $T_2$  and  $T_3$  in the collection  $\mathcal{T}_3$ .

### 5.2.2.4 Paths appended with stars

In this subsection, we analyze the identification numbers of paths appended with stars with respect to the conjectured bound. We begin by establishing the identification numbers of the trees  $T_2$  and  $T_3$  in the collection  $\mathcal{T}_3$  illustrated in Figures 5.2(c) and 5.2(d), respectively.

**Lemma 5.8.** *If  $T_2$  is the tree in  $\mathcal{T}_3$  of order  $n = 7$  and of maximum degree  $\Delta = 3$ , then*

$$\gamma^{\text{ID}}(T_2) = 5 = \frac{2}{3} \times 7 + \frac{1}{3} = \left( \frac{\Delta - 1}{\Delta} \right) n + \frac{1}{3}.$$

*Proof.* Assume that  $T_2 = P \triangleright_z S_1$ , where  $P$  is a 4-path,  $z$  is a leaf of  $P$  and  $S_1$  is a 3-star (see Figure 5.2(c)). Assume that  $V(P) = \{w, x, y, z\}$ , where  $w$  and  $z$  are the leaves of  $P$  with support vertices  $x$  and  $y$ , respectively. Further, assume that  $V(S_1) = \{u_1, a_1, b_1, c_1\}$ , where  $u_1$  is the universal vertex of  $S_1$  and the vertices  $a_1, b_1, c_1$  are the three leaves of  $S_1$ . Also assume that  $S_1$  is appended by its leaf  $c_1$  onto the leaf  $z$  of  $P$ .

We first show that any identifying code of  $T_2$  has order at least 5. So, let  $C$  be an identifying code of  $T_2$ . Then  $C$  must contain at least two vertices each from the sets  $W_0 = \{w, x, y\}$  and  $W_1 = \{u_1, a_1, b_1\}$ . Now, if either of  $W_0$  or  $W_1$  is a subset of  $C$ , then we are done. So, assume that exactly two vertices from each of  $W_0$  and  $W_1$  belong to  $C$ . Moreover, if  $z \in C$ , then we are done again. So, we also assume that  $z \notin C$ . We must have  $y \in C$  as the unique separating  $C$ -codeword for the pair  $x$  and  $w$  in  $G$ . Moreover, for  $C$  to dominate  $w$ , we need  $C \cap \{w, x\} \neq \emptyset$ . This implies that  $w \in C$ , or else, the pair  $x$  and  $y$  are not closed-separated by  $C$  in  $G$  (notice that  $z \notin C$  by assumption). Therefore,  $C \cap W_0 = \{w, y\}$ . Further, for  $C$  to separate  $a_1$  and  $b_1$ , at least one of them must be in  $C$ . So, without loss of generality, let us assume that  $a_1 \in C$ . Then, for  $C$  to dominate  $b_1$ , we need  $C \cap \{b_1, u_1\} \neq \emptyset$ . Now, for  $C$  to separate  $y$  and  $z$  in  $G$ , we must have  $u_1 \in C$  (notice that  $x \notin C$ ). This implies that  $b_1 \notin C$  (since  $|C \cap W_1| = 2$  by our assumption). However, this implies that  $C$  does not separate  $u_1$  and  $a_1$  in  $G$ , a contradiction. Hence,  $|C| \geq 5$ .

As for proving that the identifying number of  $T_2$  is bounded above by 5, since  $\gamma^{\text{ID}}(P) = 3 = \frac{2}{3} \times 4 + \frac{1}{3}$ , taking  $G_0 = P$  and  $c = \frac{1}{3}$  in Lemma 5.7, we have  $\gamma^{\text{ID}}(T_2) \leq \frac{2}{3} \times 7 + \frac{1}{3} = 5$ .  $\square$

**Lemma 5.9.** *If  $T_3$  is the tree in  $\mathcal{T}_3$  of order  $n = 10$  and of maximum degree  $\Delta = 3$ , then*

$$\gamma^{\text{ID}}(T_3) = 7 = \frac{2}{3} \times 10 + \frac{1}{3} = \left( \frac{\Delta - 1}{\Delta} \right) n + \frac{1}{3}.$$

*Proof.* Assume that  $T_2 = P \triangleright_z S_1$  and  $T_3 = T_2 \triangleright_z S_2$ , where  $P$  is a 4-path,  $z$  is a leaf of  $P$  and  $S_1$  and  $S_2$  are 3-stars (see Figure 5.2(d)). Assume that  $V(P) = \{w, x, y, z\}$ , where  $w$  and  $z$  are the leaves of  $P$  with support vertices  $x$  and  $y$ , respectively. Further, assume that, for  $i \in [2]$ ,  $V(S_i) = \{u_i, a_i, b_i, c_i\}$ , where  $u_i$  is the universal vertex of  $S_i$  and  $a_i, b_i, c_i$  are the three leaves of  $S_i$ . Assume that each  $S_i$  is appended by its leaf  $c_i$  onto the leaf  $z$  of  $P$ . We first show that any identifying code of  $T_3$  has order at least 7. So, let  $C$  be an identifying code of  $T_3$ . As in the proof of Lemma 5.8,  $C$  must contain at least two vertices from each of the sets  $W_0 = \{w, x, y\}$  and  $W_i = \{u_i, a_i, b_i\}$  for  $i \in [2]$ . Now, if any of  $W_0, W_1$  and  $W_2$  are subsets of  $C$ , then we are done. So, assume that exactly two vertices from each of  $W_0, W_1$  and  $W_2$  belong to  $C$ . Moreover, if  $z \in C$ , then we are done again. So, we also assume that  $z \notin C$ . Then, for both  $i \in [2]$ ,  $u_i \notin C$  (or else, exactly one of  $a_i$  and  $b_i$  belongs to  $C$ ; and thus, if  $a_i \in C$ , without loss of generality, then  $z$  is forced to be in  $C$  as the only separating  $C$ -codeword for the pair  $u_i, a_i$  in  $G$ , contradicting our assumption that  $z \notin C$ ).

On the other hand, we know that the vertex  $y$  must belong to  $C$  as the only separating  $C$ -codeword for the pair  $w, x$  in  $G$ . Since, by our assumption,  $C$  contains exactly two vertices from  $W_0$ , this

forces exactly one of  $w$  and  $x$  to belong to the set  $C$ . Thus, if  $w \notin C$ , then the pair  $x, y$  has no separating  $C$ -codeword in  $G$  (since, by our assumption,  $z \notin C$  either); and if  $x \notin C$ , then the pair  $y, z$  has no separating  $C$ -codeword in  $G$  (since, again, for both  $i \in [2]$ ,  $u_i \notin C$ ). Either way, we produce a contradiction. Hence,  $|C| \geq 7$ .

As for proving that the identifying number of  $T_3$  is bounded above by 7, since  $\gamma^{\text{ID}}(T_2) = 5 = \frac{2}{3} \times 7 + \frac{1}{3}$ , taking  $G_0 = T_2$  and  $c = \frac{1}{3}$  in Lemma 5.7, we have  $\gamma^{\text{ID}}(T_3) \leq \frac{2}{3} \times 10 + \frac{1}{3} = 7$ .  $\square$

Lemmas 5.8 and 5.9 therefore establish the result in Theorem 5.6 for the trees  $T_2$  and  $T_3$  in the collection  $\mathcal{T}_3$ . For the rest of this subsection, we look at other paths appended with stars which are *not* isomorphic to either  $T_2$  or  $T_3$  and show that they satisfy Conjecture 5.2 with constant  $c = 0$ .

**Lemma 5.10.** *Let  $G = P \triangleright_v S$  be a graph of order  $n$ , where  $S$  is a  $\Delta$ -star with  $\Delta \geq 3$ ,  $P$  is a path and  $v$  is a vertex of  $P$ . If any one of the following properties hold*

- (1)  *$P$  is not a 4-path; or*
- (2)  *$P$  is a 4-path and  $v$  is a non-leaf vertex of  $P$ ; or*
- (3)  *$P$  is a 4-path,  $v$  is a leaf of  $P$  and  $\Delta \geq 4$ ,*

*then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .*

*Proof.* The maximum degree of  $G$  is  $\Delta$ . Let  $P$  be an  $m$ -path and  $v$  denote the vertex of  $P$  to which the star  $S$  is appended. Let  $l_1, l_2, \dots, l_{\Delta-1}$  be the leaves that  $G$  inherits from  $S$  and let  $u$  be the universal vertex of  $S$ , that is,  $u$  is the common support vertex of the leaves  $l_1, l_2, \dots, l_{\Delta-1}$  of  $G$ . We note that  $n = m + \Delta$ . We divide the proof into cases each dealing with a type given in the statement. Furthermore, let  $v$  be the vertex of  $P$  adjacent to  $u$  and (if they exist) the other vertices of  $P$  are called, from the closest vertex to  $u$  to the farthest,  $y, x$  and  $w$ .

► **Case 7:  $P$  is not a 4-path.**

We further divide this case into the following subcases.

►► **Case 7.1:  $m = 2$ .**

In this case  $\{u, v\}$  is a dominating set in  $P \triangleright_v S$ . By Theorem 5.7, we have

$$\gamma^{\text{ID}}(P \triangleright_v S) \leq \Delta + 2 - 2 = \Delta < \left(\frac{\Delta-1}{\Delta}\right)(\Delta+2) = \left(\frac{\Delta-1}{\Delta}\right)n.$$

See Figure 5.3a for an illustration of an identifying code in  $P \triangleright_v S$ . ◀◀

►► **Case 7.2:  $m \geq 3$  and  $m \neq 4$ .**

Since  $n = m + \Delta$ , we have  $\left(\frac{\Delta-1}{\Delta}\right)n = m + \Delta - \frac{m}{\Delta} - 1$ . By supposition  $\Delta \geq 3$ . When  $m \geq 3$ , we have  $\frac{m-1}{2} \geq \frac{m}{3} \geq \frac{m}{\Delta}$  and when  $m \geq 6$ , we have  $\frac{m-2}{2} \geq \frac{m}{3} \geq \frac{m}{\Delta}$ . Let us first consider odd  $m \geq 3$ . Using Lemma 5.6 and Theorem 5.4, we therefore infer that

$$\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(P) + \Delta - 1 = \frac{m+1}{2} + \Delta - 1 = m + \Delta - 1 - \frac{m-1}{2} \leq m + \Delta - 1 - \frac{m}{\Delta} = \left(\frac{\Delta-1}{\Delta}\right)n.$$

We now consider even  $m$ . Since  $m \neq 4$ , we have  $m \geq 6$  and hence,

$$\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(P) + \Delta - 1 = \frac{m+2}{2} + \Delta - 1 = m + \Delta - 1 - \frac{m-2}{2} \leq m + \Delta - 1 - \frac{m}{\Delta} = \left(\frac{\Delta-1}{\Delta}\right)n.$$

◀◀

◀

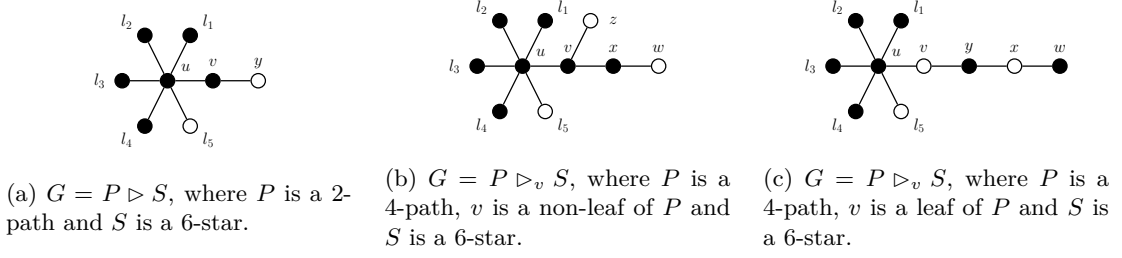


Figure 5.3: Examples for the cases in the proof of Lemma 5.10. The black vertices in each graph  $G$  constitute an identifying code.

► **Case 8:  $P$  is a 4-path and  $S$  is appended onto one of the non-leaf vertices of  $P$ .**

In this case,  $v$  is a non-leaf vertex of  $P$ . Now, let  $x$  be the (other) non-leaf vertex of  $P$  adjacent to  $v$ . The set  $C = \{u, v, x, l_1, l_2, \dots, l_{\Delta-2}\}$  is an identifying code of  $G$  as illustrated in Figure 5.3b. Since the identifying code  $C$  has cardinality

$$|C| = \Delta + 1 < \left( \frac{\Delta - 1}{\Delta} \right) (\Delta + 4) = \left( \frac{\Delta - 1}{\Delta} \right) n,$$

the desired result follows. ◀

► **Case 9:  $P$  is a 4-path,  $S$  is appended onto a leaf of  $P$  and  $\Delta \geq 4$ .**

In this case,  $v$  is a leaf of  $P$ . The set  $C = \{u, w, y, l_1, l_2, \dots, l_{\Delta-2}\}$  is an identifying code of  $G$  as illustrated in Figure 5.3c. Since the identifying code  $C$  has cardinality

$$|C| = \Delta + 1 < \left( \frac{\Delta - 1}{\Delta} \right) (\Delta + 4) = \left( \frac{\Delta - 1}{\Delta} \right) n,$$

the desired result follows. ◀

This proves the result. ◻

Lemma 5.10 shows that all trees of the form  $P \triangleright S$ , where  $P$  is a 4-path and  $S$  is a star, but not isomorphic to  $T_2$ , satisfy the bound in Conjecture 5.2 with  $c = 0$ . The next corollary follows immediately from Lemma 5.10.

**Corollary 5.3.** *If  $G = P \triangleright_v S$  is a graph of order  $n$ , where  $P$  is a path,  $v$  is a vertex of  $P$  and  $S$  is a  $\Delta$ -star with  $\Delta \geq 4$ , then*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n.$$

Our next lemma shows that all trees of the form  $T_2 \triangleright S$ , where  $S$  is a star, but not isomorphic to  $T_3$  satisfy the bound in Conjecture 5.2 with  $c = 0$ .

**Lemma 5.11.** *If  $G = T_2 \triangleright_v S$  is a graph of order  $n$  and of maximum degree  $\Delta \geq 3$  such that  $G \not\cong T_3$ , where  $v$  is a vertex of  $T_2$  and  $S$  is a  $\Delta_S$ -star with  $\Delta_S \geq 3$ , then*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n.$$

*Proof.* Clearly,  $\Delta_S \leq \Delta$ . We consider the following cases.

► **Case 10:  $\Delta \geq 4$ .**

Observe in this case that either  $\Delta_S = \Delta$  or  $\Delta = 4$  and  $\Delta_S = 3$ . By Lemma 5.8, we have  $\gamma^{\text{ID}}(T_2) = 5$ . Therefore, by Lemma 5.6 we infer that

$$\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(T_2) + \Delta_S - 1 = \Delta_S + 4 < \left( \frac{\Delta - 1}{\Delta} \right) (\Delta_S + 7) = \left( \frac{\Delta - 1}{\Delta} \right) n,$$



(a)  $G = T_2 \triangleright_{a_1} S$ , where  $S$  is a 3-star and  $a_1$  is a leaf of  $T_2$ . (b)  $G = T_2 \triangleright_w S$ , where  $S$  is a 3-star and  $w$  is a leaf of  $T_2$ .

Figure 5.4: Figures illustrating the cases in Lemma 5.11. The black vertices constitute an identifying code.

yielding the desired result in this case.

Thus, we assume from now on that  $\Delta = \Delta_S = 3$ . Hence, the graph  $G$  has order  $n = 10$  and it suffices for us to show that there exists an identifying code  $C$  of  $G$  of cardinality  $|C| = 6 < \frac{2}{3} \times 10 = (\frac{\Delta-1}{\Delta})n$ . Let  $T_2 = P \triangleright_z S_1$ , where  $P$  is a 4-path,  $z$  is a leaf of  $P$  and  $S_1$  is a  $\Delta$ -star for  $\Delta \geq 3$ . Assume further that

- (1)  $V(P) = \{w, x, y, z\}$ , where  $w$  and  $z$  are leaves of  $P$  with support vertices  $x$  and  $y$ , respectively.
- (2)  $V(S_1) = \{u_1, a_1, b_1, c_1\}$ , where  $u_1$  is the universal vertex of  $S_1$ , the vertices  $a_1, b_1, c_1$  are the three leaves of  $S_1$  and that  $S_1$  is appended by its leaf  $c_1$  onto the leaf  $z$  of  $P$ .
- (3)  $V(S) = \{u, l_1, l_2, l_3\}$ , where  $u$  is the universal vertex of  $S$ , the vertices  $l_1, l_2, l_3$  are the leaves of  $S$  and  $l_3$  is the leaf of  $S$  by which the latter is appended onto the vertex  $v$  of  $T_2$ .

Since  $\Delta = \Delta_S = 3$ ,  $S$  is not appended onto the universal vertex  $u_1$  of  $S_1$ . We consider next two further cases. ◀

► **Case 11:  $S$  is appended onto either of the leaves  $a_1$  and  $b_1$  of  $S_1$ .**

Without loss of generality, let us assume that  $S$  is appended onto the leaf  $a_1$  of  $T_2$ . The set  $C = \{y, w, u_1, u, a_1, l_1\}$  is an identifying code of  $G$  of cardinality 6 as illustrated in Figure 5.4a, where the vertices in the identifying code are marked with black, implying that  $\gamma^{\text{ID}}(G) \leq 6 < \frac{2}{3} \times 10 = (\frac{\Delta-1}{\Delta})n$ , yielding the desired result. ◀

► **Case 12:  $S$  is appended onto any vertex of  $T_2$  other than  $u_1, a_1$  and  $b_1$ .**

In this case, since  $G \not\cong T_3$ , the star  $S$  cannot be appended onto the vertex  $z$  of  $T_2$  (that is,  $v \neq z$ ). Thus,  $S$  is appended onto either of the vertices  $w, x$  and  $y$  of  $T_2$ . We further divide this case into the following subcases.

►► **Case 12.1:  $S$  is appended onto either of the non-leaf vertices  $x$  or  $y$  of  $T_2$ .**

In this case, the graph  $G$  can also be expressed as  $G = G' \triangleright_z S_1$ , where  $G' = P \triangleright_v S$ . Since the maximum degree of  $G'$  is  $\Delta_S = 3$ , by Lemma 5.10(2), we have

$$\gamma^{\text{ID}}(G') \leq \frac{2}{3}|V(G')| = \frac{2}{3}(n-3).$$

The bound for  $G$  therefore follows with Lemma 5.6. ◀◀

►► **Case 12.2:  $S$  is appended onto the leaf  $w$  of  $T_2$ .**

In this case, the set  $C = \{a_1, u_1, z, w, u, l_1\}$  is an identifying code of  $G$  of cardinality 6 as illustrated in Figure 5.4b where the vertices in the identifying code are marked with black, implying that  $\gamma^{\text{ID}}(G) \leq 6 < (\frac{\Delta-1}{\Delta})n$ , as desired. ◀◀

Hence, the result holds in this case. ◀

This completes the proof. ◻

Finally, the next lemma shows that all trees of the form  $T_3 \triangleright S$ , where  $S$  is a star, also satisfy the bound in Conjecture 5.2 with  $c = 0$ .

**Lemma 5.12.** *If  $G = T_3 \triangleright_v S$  is a graph of order  $n$  and of maximum degree  $\Delta \geq 3$ , where  $v$  is a vertex of  $T_3$  and  $S$  is a  $\Delta_S$ -star with  $\Delta_S \geq 3$ , then*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n.$$

*Proof.* Let  $P$  be a 4-path and let  $S_1$  and  $S_2$  be 3-stars such that  $V(P) = \{w, x, y, z\}$ , where  $w$  and  $z$  are leaves of  $P$  with support vertices  $x$  and  $y$ , respectively. Moreover, for  $i \in [2]$ , we have  $V(S_i) = \{u_i, a_i, b_i, c_i\}$ , where  $u_i$  is the universal vertex of  $S_i$ , and the vertices  $a_i, b_i, c_i$  are the three leaves of  $S_i$ . Also assume that each  $S_i$  is appended by its leaf  $c_i$  onto the leaf  $z$  of  $P$  to form the tree  $T_3$ . Now, let  $T_2 \cong P \triangleright_z S_1$  and we have  $T_3 \cong (P \triangleright_z S_1) \triangleright_z S_2$ . Let us assume first that  $S$  is also appended onto the vertex  $z$  of  $P$  (that is,  $v = z$ ). In this case,  $\deg_G(z) = 4$ . If  $\Delta_S = 3$ , then  $n = 13$  and by Lemma 5.6, we have

$$\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(T_3) + 2 = 9 < \frac{3}{4} \times 13 = \left( \frac{\Delta - 1}{\Delta} \right) n.$$

This implies that  $S$  is a  $\Delta$ -star with  $\Delta \geq 4$ . Let  $G' = P \triangleright_z S$  have order  $n' (= n - 6)$  and maximum degree  $\Delta' \leq \Delta$ . By Corollary 5.3, we have

$$\gamma^{\text{ID}}(G') \leq \left( \frac{\Delta' - 1}{\Delta'} \right) n' \leq \left( \frac{\Delta - 1}{\Delta} \right) (n - 6).$$

Therefore, by Lemma 5.7, we have the result.

So, let us assume that  $G'' = T_2 \triangleright_v S$  has order  $n'' (= n - 3)$  and maximum degree  $\Delta'' \leq \Delta$ , where  $S$  is appended onto a vertex of  $T_2$  other than  $z$  (that is,  $v \neq z$ ). In that case, the graph  $G'' \not\cong T_3$  and, therefore, by Lemma 5.11, we have

$$\gamma^{\text{ID}}(G'') \leq \left( \frac{\Delta'' - 1}{\Delta''} \right) n'' \leq \left( \frac{\Delta - 1}{\Delta} \right) (n - 3).$$

Therefore, we have the result by Lemma 5.7. □

To conclude the subsection, we close with the following remark.

**Remark 5.4.** *If  $G$  is a graph isomorphic to a path appended with stars, then  $G$  satisfies Conjecture 5.2 with  $c = \frac{1}{3}$  in the case of  $T_2$  and  $T_3$ , and with  $c = 0$  otherwise.*

### 5.2.2.5 Appended stars

The trees in the collection  $\mathcal{T}_3$  other than  $T_2$  and  $T_3$  are precisely the appended 3-stars of maximum degree 3 and of diameter at most 6. In this subsection, we show that all appended stars satisfy the conjectured bound with  $c = 0$ , except for those in  $\mathcal{T}_3$ , which satisfy the bound with  $c = \frac{1}{3}$ . To start with, we first look at appended stars of maximum degree at least 4.

**Lemma 5.13.** *If  $G$  is an appended star of order  $n$  and of maximum degree  $\Delta \geq 4$ , then*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n.$$

*Proof.* Let  $p$  be an integer and, for  $i \in [0, p]$ , let  $S_i$  be a  $\Delta_i$ -star with  $\Delta_i \geq 3$ . Also, for  $i \in [p]$ , let  $G_i = G_{i-1} \triangleright_{v_{i-1}} S_i$ , where  $v_{i-1}$  is a vertex of  $G_{i-1}$ , and such that  $G = G_p$  is the appended star. Let  $V(S_i) = \{u_i, l_1^i, \dots, l_{\Delta_i}^i\}$ , where  $u_i$  is the universal vertex and each  $l_j^i$ , for  $j \in [\Delta_i]$ , is a leaf of  $S_i$ . For  $h \in [p]$ , let  $G_h$  have order  $n'$  and maximum degree  $\Delta'$ . Observe that if  $\gamma^{\text{ID}}(G_h) \leq \left( \frac{\Delta' - 1}{\Delta'} \right) n'$ , then  $\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n$  by Lemma 5.7. We next look at the following two cases according to whether the  $\Delta_i$ 's are all equal or not.

► **Case 13:** There exists  $1 \leq h \leq p$  such that  $\Delta_0 = \Delta_1 = \dots = \Delta_{h-1} \neq \Delta_h$ .

Let  $G_{h-1} = G''$  have order  $n''$  and maximum degree  $\Delta''$ . Moreover, let  $C''$  be a minimum identifying code of  $G''$ .

►► **Case 13.1:**  $\Delta_h = \Delta_{h-1} + q$  for some  $q > 0$ .

Observe that

$$n'' = h\Delta_0 + 1 \text{ and } n' = n'' + \Delta_h = (h+1)\Delta_h - hq + 1.$$

By Corollary 5.2, we have

$$|C''| \leq \left( \frac{\Delta_0 - 1}{\Delta_0} \right) n'' + \frac{1}{\Delta_0} = h(\Delta_0 - 1) + 1 = h(\Delta_h - q - 1) + 1.$$

Moreover, by Lemma 5.6, we have

$$\begin{aligned} \gamma^{\text{ID}}(G_h) &\leq |C''| + \Delta_h - 1 \leq h(\Delta_h - q - 1) + 1 + (\Delta_h - 1) \\ &= \left( \frac{\Delta_h - 1}{\Delta_h} \right) h\Delta_h - hq + \Delta_h \\ &= \left( \frac{\Delta_h - 1}{\Delta_h} \right) (n' - \Delta_h + hq - 1) - hq + \Delta_h \\ &= \left( \frac{\Delta_h - 1}{\Delta_h} \right) n' - (hq - 1) \frac{1}{\Delta_h} \\ &\leq \left( \frac{\Delta' - 1}{\Delta'} \right) n'. \end{aligned}$$

◀◀

►► **Case 13.2:**  $\Delta_h = \Delta_{h-1} - q$  for some  $q > 0$ .

Observe that

$$n'' = h\Delta_0 + 1 \text{ and } n' = n'' + \Delta_h = (h+1)\Delta_0 - q + 1.$$

By Corollary 5.2, we have

$$|C''| \leq \left( \frac{\Delta_0 - 1}{\Delta_0} \right) n'' + \frac{1}{\Delta_0} = h(\Delta_0 - 1) + 1.$$

Moreover, by Lemma 5.6, we have

$$\begin{aligned} \gamma^{\text{ID}}(G_h) &\leq |C''| + \Delta_h - 1 \leq h(\Delta_0 - 1) + \Delta_0 - q \\ &= \left( \frac{\Delta_0 - 1}{\Delta_0} \right) h\Delta_0 + \Delta_0 - q \\ &= \left( \frac{\Delta_0 - 1}{\Delta_0} \right) (n' - \Delta_0 + q - 1) + \Delta_0 - q \\ &= \left( \frac{\Delta_0 - 1}{\Delta_0} \right) n' - \frac{q - 1}{\Delta_0} \\ &\leq \left( \frac{\Delta_0 - 1}{\Delta_0} \right) n' \leq \left( \frac{\Delta' - 1}{\Delta'} \right) n'. \end{aligned}$$

◀◀

In both the above subcases therefore, the result follows from Lemma 5.7. This completes the proof of Case 1. ◀

► **Case 14:**  $\Delta_0 = \Delta_1 = \dots = \Delta_p$ .

In this case, we have  $n = (p+1)\Delta_0 + 1$  and  $n' = (h+1)\Delta_0 + 1$ . Let us now divide our analysis into the following subcases.



►► **Case 14.1:** For some  $i \in [p]$ , the vertex  $v_{i-1} \in \{u_0, u_1, \dots, u_{i-1}\}$ .

This subcase implies that the star  $S_i$  is appended onto a universal vertex of any of the stars  $S_0, S_1, \dots, S_{i-1}$ . Then by a possible renaming of the stars, let us assume that  $v_0 = u_0$  is the universal vertex of  $S_0$ . We then take  $h = 1$  and look at the graph  $G_1 = S_0 \triangleright_{u_0} S_1$ . Therefore, we have  $\Delta' = \Delta_0 + 1 \geq 4$ . The set  $C'$  consisting of all the leaves of  $G_1$  is an identifying code of  $G_1$ . Therefore, we have

$$\gamma^{\text{ID}}(G_1) \leq |C'| = 2\Delta_0 - 1 = \left(\frac{\Delta_0}{\Delta_0 + 1}\right)(2\Delta_0 + 1) - \frac{1}{\Delta_0 + 1} \leq \left(\frac{\Delta' - 1}{\Delta'}\right)n'.$$

Replacing  $G_0$  by  $G_1$  in Lemma 5.7, yields the desired result. ◀◀

►► **Case 14.2:** For each  $1 \leq i \leq p$ , the star  $S_i$  is appended onto a leaf of  $G_{i-1}$ .

This subcase can be further divided according to whether  $\Delta_0 \geq 4$  or  $\Delta_0 = 3$ .

►►► **Case 14.2.1:**  $\Delta_0 \geq 4$ .

We again take  $h = 1$  and consider the graph  $G_1$  of order  $n' = 2\Delta_0 + 1$  with  $\Delta' = \Delta_0 \geq 4$ . In this case,  $G_1$  has  $2\Delta' - 2$  leaves, a single degree 2 vertex, namely  $l_1^2$ , and two degree  $\Delta'$  vertices, namely  $u_1$  and  $u_2$ . The set  $C' = \{u_1, u_2, l_2^1, \dots, l_{\Delta'-1}^1, l_2^2, \dots, l_{\Delta'-1}^2\}$  is an identifying code of  $G_1$ . Moreover, we have

$$|C'| = 2(\Delta' - 1) = \left(\frac{\Delta' - 1}{\Delta'}\right)n' - \frac{\Delta' - 1}{\Delta'} < \left(\frac{\Delta' - 1}{\Delta'}\right)n'.$$

Therefore, again replacing  $G_0$  by  $G_1$  in Lemma 5.7, the result follows. ◀◀◀

►►► **Case 14.2.2:**  $\Delta_0 = 3$ .

Since  $\Delta \geq 4$ , this implies, again by a possible renaming of the stars, that the graph  $G_3$  is obtained by appending the stars  $S_1, S_2$  and  $S_3$  onto a single leaf  $l_1^0$ , say, of  $S_0$ . The set  $C' = \{l_1^0, l_2^0, l_1^1, l_1^2, l_1^3, u_0, u_1, u_2, u_3\}$  is an identifying code of  $G_3$ . This implies that

$$|C'| = 9 < \frac{3}{4} \times 13 \leq \left(\frac{\Delta' - 1}{\Delta'}\right)n'.$$

Hence, again replacing  $G_0$  by  $G_3$  in Lemma 5.7, the result follows. ◀◀◀

This proves the result in the current case. ◀◀

Hence, the result is true when  $\Delta_0 = \Delta_1 = \dots = \Delta_p$ . ◀

This proves the result. ◻

The preceding lemma shows that if we require  $c > 0$  in the conjectured bound for an appended star, then the appended star must be subcubic. For the rest of this subsection, therefore, we investigate the veracity of Conjecture 5.2 for appended 3-stars of maximum degree 3. Before that, one can verify the following.

**Remark 5.5.** *The diameter of a subcubic appended 3-star is even.*

We now look at the appended 3-stars in  $\mathcal{T}_3$  in the following proposition, which proves that the trees in  $\mathcal{T}_3$  satisfy the conjectured bound with  $c = \frac{1}{3}$ .

**Proposition 5.2.** *If  $G$  is an appended 3-star of order  $n$ , of maximum degree  $\Delta = 3$  and of diameter at most 6, then*

$$\gamma^{\text{ID}}(G) = \frac{2}{3}n + \frac{1}{3}.$$

*Proof.* For  $\text{diam}(G) = 2$ , that is,  $G$  being a 3-star itself, the result follows from Lemma 5.5. We therefore consider  $\text{diam}(G) > 2$ , that is, by the parity of  $\text{diam}(G)$  by Remark 5.5, only the cases where  $\text{diam}(G)$  is either 4 or 6. We first show that  $\gamma^{\text{ID}}(G) \geq \frac{2}{3}n + \frac{1}{3}$ . Let  $p$  be an integer and for each  $i \in [0, p]$ , let  $S_i$  be a 3-star. Also, for  $i \in [p]$ , let  $G_i = G_{i-1} \triangleright_{v_{i-1}} S_i$ , where  $v_{i-1}$  is a vertex of  $G_{i-1}$ , and where  $G = G_p$  is the appended 3-star.

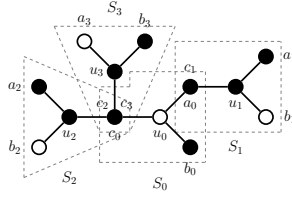


Figure 5.5: Tree  $T_7 \in \mathcal{T}_3$ :  $\text{diam}(T_7) = 6$  and  $\gamma^{\text{ID}}(T_7) = 9$ . The set of black vertices constitutes an identifying code of  $T_7$ .

Suppose, to the contrary, that  $G$  (with diameter either 4 or 6) is a counterexample of minimum order such that  $\gamma^{\text{ID}}(G) < \frac{2}{3}n + \frac{1}{3}$ . Since  $\text{diam}(G)$  is either 4 or 6, there exists a 3-star  $S_0$ , say, such that  $G$  is obtained by appending other 3-stars to the leaves of  $S_0$  only (notice that, for  $G$  to be subcubic, no 3-star  $S_j$  is appended onto the universal vertex of another 3-star  $S_i$ ). For each  $i \in [0, p]$ , let  $V(S_i) = \{u_i, a_i, b_i, c_i\}$ , where  $u_i$  is the universal vertex and  $a_i, b_i, c_i$  are the leaves of  $S_i$ . Moreover, assume that each  $S_i$  is appended by its leaf  $c_i$  onto the graph  $G_{i-1}$ . Then, for all  $i \in [p]$ ,  $a_i$  and  $b_i$  are leaves of  $G$  adjacent to the common support vertex  $u_i$  (see Figure 5.5 for the tree  $T_7 \in \mathcal{T}_3$  as a sample appended 3-star of diameter at most 6; also refer to Figure 5.2(a) and Figures 5.2(d)-5.2(k) for other such examples). Now, assume  $C$  to be a minimum identifying code of  $G$ . Then, we claim the following.

■ **Claim 1.** For each  $i \in [p]$ , the universal vertex  $u_i$  of the 3-star  $S_i$  belongs to  $C$ .

*Proof of claim.* Suppose, to the contrary, that  $u_i \notin C$  for some  $i \in [p]$ . For the vertices  $a_i$  and  $b_i$ , both of which have degree 1 in  $G$ , to be dominated by  $C$ , we must have  $a_i, b_i \in C$ . Now, observe that, by a possible renaming of the stars  $S_1, S_2, \dots, S_p$ , we can assume that  $u_p \notin C$  and that  $a_p, b_p \in C$ . Let  $C' = C \setminus \{a_p, b_p\}$ . Then,  $C'$  is an identifying code of  $G_{p-1}$ . Noting that  $n = 3p + 4$  and  $|C| < \frac{2}{3}n + \frac{1}{3}$ , we have

$$\begin{aligned} |C'| &= |C| - 2 < \frac{2}{3}n - \frac{5}{3} = \frac{2}{3}(3p + 4) - \frac{5}{3} = 2p + 1 \\ \implies |C'| &\leq 2p < 2p + \frac{2}{3} = \frac{2}{3}(3p + 1) = \frac{2}{3}(n - 3). \end{aligned}$$

However, since  $G_{p-1}$  has order  $n - 3$ , this contradicts the minimality of the order of  $G$ . Thus, we must have  $u_i \in C$  for all  $i \in [p]$ . ■

Claim 1 implies that, for each  $i \in [p]$ , at least two vertices from each set  $\{a_i, b_i, c_i\}$ , in addition to  $u_i$ , must belong to  $C$  (otherwise the vertices of  $S_i$  are not separated). However, if both  $a_i, b_i \in C$ , then we can discard  $b_i$  from  $C$  and include  $c_i$  (if not included already) in the identifying code  $C$  of  $G$ . In this way,  $C$  still remains an identifying code of  $G$  and of order at most the same as before. In particular then, for each  $i \in [p]$ , we can assume that  $\{u_i, a_i, c_i\} \subset C$ . Next, we prove the following claim.

■ **Claim 2.**  $|C| \geq 2p + 3$ .

*Proof of claim.* If  $\text{diam}(G) = 4$ , then in this case,  $p \leq 2$  and onto *exactly one* leaf  $c_0$ , say, of  $S_0$  at least one other 3-star  $S_1$ , say, is appended by its leaf  $c_1$ . However then at least two vertices from  $\{u_0, a_0, b_0\}$  must be in  $C$ . By the above considerations following the proof of Claim 1, we have at least  $2p + 1$  further vertices in  $C$ , and thus  $|C| \geq 2p + 3$ . Hence, we may assume that  $\text{diam}(G) = 6$ , for otherwise the desired lower bound in the claim follows. In this case,  $p \leq 6$  and to *at least two* leaves  $a_0$  and  $c_0$  of  $S_0$ , other 3-stars are appended. Then again, by our previous observation,  $a_0$  and  $c_0$  are included in the identifying code  $C$ , since  $a_0 = c_i$  for some  $i \neq 0$  and  $c_0 = c_j$  for some  $j \neq 0$ . Moreover, for  $b_0$  to be dominated by  $C$ , at least one of  $\{u_0, b_0\}$  must be in  $C$ . Thus, we again have

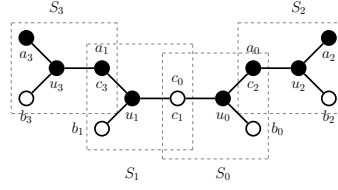


Figure 5.6: Graph  $Z$ :  $\gamma^{\text{ID}}(Z) = 8$ . The set of black vertices constitutes an identifying code of  $Z$ .

$|C| \geq 2p + 3$ . This proves the claim.  $\blacksquare$

Finally, again noting that  $n = 3p + 4$ , we have by Claim 2 that

$$\gamma^{\text{ID}}(G) = |C| \geq 2p + 3 = \frac{2}{3}(3p + 3) + 1 = \frac{2}{3}n + \frac{1}{3},$$

contradicting our initial supposition that  $\gamma^{\text{ID}}(G) < \frac{2}{3}n + \frac{1}{3}$ . Hence,  $\gamma^{\text{ID}}(G) \geq \frac{2}{3}n + \frac{1}{3}$ . On the other hand, by Corollary 5.2, we have  $\gamma^{\text{ID}}(G) \leq \frac{2}{3}n + \frac{1}{3}$ . Consequently,  $\gamma^{\text{ID}}(G) = \frac{2}{3}n + \frac{1}{3}$ .  $\square$

We now turn our attention to the appended 3-stars of diameter strictly larger than 6, that is, at least 8 (by the parity of the diameter in Remark 5.5). To that end, we first show in the next lemma the existence of a special appended 3-star of diameter 8.

**Lemma 5.14.** *Let each of  $S_0, S_1, S_2, S_3$  be a 3-star. Let  $G_0 = S_0$  and, for  $i \in [3]$ , let  $G_i = G_{i-1} \triangleright_{v_{i-1}} S_i$ , where  $v_{i-1}$  is a vertex of  $G_{i-1}$ . Finally, let  $Z = G_3$  be a subcubic appended 3-star of diameter 8. Then, up to isomorphism, the graph  $Z$  is unique.*

*Proof.* Since  $Z = G_2 \triangleright_{v_2} S_3$  and  $\text{diam}(Z) = 8$ , it follows that  $\text{diam}(G_2) = 6$ . Next, to obtain  $G_2$ , assume, without loss of generality, that  $S_1$  and  $S_2$  are appended onto two distinct leaves of  $S_0$ . Then, the longest path in  $G_2$  (with path-length of  $\text{diam}(G_2)$ ) has one leaf each of  $S_1$  and  $S_2$  as its endpoints. Thus,  $S_3$  is appended onto a leaf of  $G_2$  that is also a leaf of either  $S_1$  or  $S_2$  (see Figure 5.6). This completes the proof.  $\square$

In the following lemma we establish the identification number of the graph  $Z$  constructed in the statement of Lemma 5.14.

**Lemma 5.15.** *If  $Z$  is the unique subcubic appended 3-star of diameter 8 as defined in the statement of Lemma 5.14, then  $\gamma^{\text{ID}}(Z) = 8 < \frac{2}{3}|V(Z)|$ .*

*Proof.* For  $0 \leq i \leq 3$ , let  $V(S_i) = \{u_i, a_i, b_i, c_i\}$  such that  $u_i$  is the universal vertex and  $a_i, b_i, c_i$  are the three leaves of  $S_i$ . To obtain  $Z$  (up to isomorphism), let us assume that the 3-stars are appended in such a manner that the following pairs of vertices are identified:  $(c_0, c_1)$ ,  $(a_0, c_2)$  and  $(a_1, c_3)$  (see Figure 5.6). To first show that  $\gamma^{\text{ID}}(Z) \leq 8$ , it is enough to check that the set  $\{u_0, u_1, u_2, u_3, a_0, a_1, a_2, a_3\}$  is an identifying code of  $Z$  (again, see Figure 5.6 for the identifying code represented by the black vertices). We next show the reverse that  $\gamma^{\text{ID}}(Z) \geq 8$ . Let  $C$  be a minimum-ordered identifying code in  $Z$ .

$\blacksquare$  **Claim 1.** If  $c_0 \notin C$ , then  $|C| \geq 8$ .

*Proof of claim.* Let  $G' = G - \{c_0\}$ . Observe that  $G'$  consists of two identical components of order 6. We show that each of them has at least four vertices in the identifying code. First of all,  $a_0 \in C$  is forced as the only vertex which separates  $u_0$  and  $b_0$ . Moreover,  $|\{u_0, b_0\} \cap C| \geq 1$  since  $C$  dominates  $b_0$ . Finally, we have  $|\{a_2, u_2, b_2\} \cap C| \geq 2$  to dominate and separate  $a_2$  and  $b_2$ . Thus,  $|C| \geq 8$ .  $\blacksquare$

■ **Claim 2.** If  $c_0 \in C$ , then  $|C| \geq 9$ .

*Proof of claim.* We again show that  $|C \cap \{u_0, b_0, a_0, u_2, a_2, b_2\}| \geq 4$  and the claim follows from symmetry. If  $u_2 \notin C$ , then in this case, we need  $a_2, b_2 \in C$  to dominate them and we also need two vertices from  $\{a_0, u_0, b_0\}$ . Indeed, a single vertex can dominate all of these three vertices only if it is  $u_0$ . However,  $u_0$  does not separate  $a_0$  and  $b_0$ , implying that  $|C| \geq 9$ . On the other hand, if  $u_2 \in C$ , then in this case, we have  $a_2$  or  $b_2$  in  $C$ . We may assume without loss of generality that  $a_2 \in C$ . To separate  $u_2$  and  $a_2$ , we also need  $a_0$  or  $b_2$  in  $C$ . Thus,  $|C \cap \{a_2, b_2, u_2, a_0\}| \geq 3$ . Finally, we need  $u_0$  or  $b_0$  in  $C$  to dominate  $b_0$ , once again implying that  $|C| \geq 9$ . ■

The lemma follows from these two claims. □

The above lemma shows that the graph  $Z$  satisfies Conjecture 5.2 with  $c = 0$ . We end this section with the following lemma which shows that all appended 3-stars of maximum degree 3 other than those in the collection  $\mathcal{T}_3$  (that is, with diameter at least 8) satisfy the conjectured bound with  $c = 0$ .

**Lemma 5.16.** *If  $G$  is an appended star of order  $n$ , of maximum degree  $\Delta = 3$  and of diameter at least 8, then*

$$\gamma^{\text{ID}}(G) \leq \frac{2}{3}n = \left(\frac{\Delta-1}{\Delta}\right)n.$$

*Proof.* Since  $\text{diam}(G) \geq 8$ , the graph  $G$  must contain the graph  $Z$  (of Lemma 5.15) as an induced subgraph. Taking  $G_0 = Z$  in Lemma 5.7, the result follows. □

We summarize the current subsection with the following remark.

**Remark 5.6.** *If  $G$  is an appended star, then  $G$  satisfies Conjecture 5.2 with  $c = \frac{1}{3}$  if  $G$  is isomorphic to a tree in  $\mathcal{T}_3$  and with  $c = 0$  otherwise.*

We also state the following proposition, which will be useful in our proofs.

**Proposition 5.3.** *If  $T$  is a tree in  $\mathcal{T}_3$ , then the following properties hold.*

- (i) *If  $T \neq T_2$ , then  $T$  has an optimal identifying code  $C(T)$  containing all vertices of degree at most 2.*
- (ii) *If  $T \notin \{T_2, T_3\}$ , then  $C(T)$  can be chosen as an independent set which contains every vertex of degree at most 2. When we delete any code vertex  $v$  from  $T$ , set  $C(T) \setminus \{v\}$  forms an optimal identifying code of the forest  $T - v$ .*

*Proof.* The constructions in the claim can be confirmed by the identifying codes provided in Figure 5.2. Their optimality follows from Lemmas 5.5, 5.8, 5.9 and Proposition 5.2. □

### 5.2.2.6 Proof of Theorem 5.6

We are now ready to prove Theorem 5.6. The proof first utilizes Lemmas 5.3 and 5.4, using which, we can show that one needs to consider only the trees  $G$  of the form  $G' \triangleright_v S$ , whereby  $G'$  is necessarily a tree as well. Induction plays a central role in proving Theorem 5.6.

*Proof of Theorem 5.6.* The proof is by induction on the order  $n$  of the tree  $G$ . Since we have  $\Delta \geq 3$ , this implies that  $n \geq 4$ . However,  $n = 4$  implies that  $G$  is a 3-star and thus is isomorphic to a graph in  $\mathcal{T}_3$ . Therefore, we take the base case of the induction hypothesis to be when  $n = 5$ . In the base case now, for  $G$  to be a tree and non-isomorphic to a star, we must have  $G \cong P \triangleright S$ , where  $P$  is a 2-path and  $S$  is a 3-star. Therefore, by Lemma 5.10(1), the result is true in the base case. Let us assume that the induction hypothesis is true for all trees  $G''$  on  $n''$  vertices such that  $5 \leq n'' < n$ , not isomorphic to a tree in  $\mathcal{T}_{\Delta''}$ , of maximum degree  $\Delta'' \geq 3$ . Let  $\ell$  and  $s$  be the number of leaves and support vertices, respectively, in  $G$ . First of all, if  $s \geq \frac{n}{\Delta}$ , then, by Lemma 5.3, we have

$$\gamma^{\text{ID}}(G) \leq n - s \leq n - \frac{1}{\Delta}n = \left(\frac{\Delta-1}{\Delta}\right)n$$

and, hence, we are done. Moreover, if  $\ell \leq (\frac{\Delta-2}{\Delta})n$ , then, by Lemma 5.4, we have

$$\gamma^{\text{ID}}(G) \leq \frac{n+\ell}{2} \leq \left(1 + \frac{\Delta-2}{\Delta}\right) \frac{n}{2} = \left(\frac{\Delta-1}{\Delta}\right) n$$

and we are done in this case too. We therefore assume that both  $s < \frac{n}{\Delta}$  and that  $\ell > \frac{\Delta-2}{\Delta}n$ . The latter inequality implies that there is at least one leaf and, hence, at least one support vertex as well in  $G$ . In this case, the average number,  $\frac{\ell}{s}$ , of leaves per support vertex satisfies  $\frac{\ell}{s} > \Delta - 2$ . Moreover, as  $G$  is not isomorphic to a star, the maximum number of leaves adjacent to a support vertex is  $\Delta - 1$ . Hence, there exists at least one support vertex which is adjacent to exactly  $\Delta - 1$  leaves. Therefore, we must have  $G \cong G' \triangleright_x S$ , where  $G'$  is also a tree,  $x$  is a vertex of  $G'$  and  $S$  is a  $\Delta$ -star. Let  $n' = |V(G')|$  and  $\Delta'$  be the maximum degree of  $G'$ . We proceed further with the following claim.

■ **Claim 1.** If  $n' \leq 4$ , then  $\gamma^{\text{ID}}(G) \leq (\frac{\Delta-1}{\Delta})n$ .

*Proof of claim.* Suppose that  $n' \leq 4$ . If  $n' = 1$ , then  $G$  is a  $\Delta$ -star by itself and, hence,  $G$  is isomorphic to a graph in  $\mathcal{T}_\Delta$ , a contradiction. If  $n' = 2$ , then since  $G'$  is connected, we have  $G' \cong P_2$ . Therefore,  $G \cong P \triangleright S$ , where  $P$  is a 2-path and  $S$  is a  $\Delta$ -star, and the desired upper bound follows by Lemma 5.10(1). If  $n' = 3$ , then since  $G'$  is a tree,  $G' \not\cong K_3$  and, thus,  $G \in P \triangleright S$ , where  $P$  is a 3-path, and once again, by Lemma 5.10(1), the desired result holds. Hence, we may assume that  $n' = 4$ .

Since  $G'$  is a tree,  $G'$  can be isomorphic to either a 4-path or a 3-star. Let us, therefore, assume that  $G'$  is isomorphic to either a 4-path  $P$ , say, or a 3-star  $S_1$ , say. If  $G' \cong P$ , then by the fact that  $G$  is not isomorphic to the graph  $T_2$  in  $\mathcal{T}_3$ , we must have  $G \cong P \triangleright_x S$ , where  $x$  is a non-leaf vertex of  $P$ . In this case, by Lemma 5.10(2), the result holds. If, however,  $G' \cong S_1$ , then  $G \in S_1 \triangleright_x S$ , where  $x$  is a vertex of  $S_1$ . If  $x$  is a leaf of  $S_1$  and  $\Delta = 3$ , then  $G \cong T_1$  in the collection  $\mathcal{T}_3$ , a contradiction. Therefore, in the case that either  $x$  is a non-leaf vertex of  $S_1$  or that  $\Delta \geq 4$ , we are done by Lemma 5.13. ■

By Claim 1, we may assume that  $n' \geq 5$ , for otherwise the desired result follows. Now, if  $\Delta' = 2$ , then  $G'$  is a path of order at least 5 and hence, we are done by Lemma 5.10(1). Hence we may assume in what follows that  $\Delta' \geq 3$ .

■ **Claim 2.** If  $G'$  is isomorphic to a graph in  $\mathcal{T}_{\Delta'}$ , then  $\gamma^{\text{ID}}(G) \leq (\frac{\Delta-1}{\Delta})n$ .

*Proof of claim.* Suppose that  $G'$  is isomorphic to a graph in  $\mathcal{T}_{\Delta'}$ . If  $G'$  is isomorphic to a star in  $\mathcal{T}_{\Delta'}$ , then  $G' \cong S'$ , where  $S'$  is a  $\Delta'$ -star. Since  $n' \geq 5$ , we must have  $\Delta' \geq 4$ , implying that  $G \cong S' \triangleright S$ , and the desired upper bound follows from Lemma 5.13. If  $G'$  is isomorphic to  $T_2$ , then  $G = T_2 \triangleright S$ . However, as  $G \not\cong T_3 \in \mathcal{T}_3$ , the result follows by Lemma 5.11. If  $G'$  is isomorphic to  $T_3$ , then in this case,  $G = T_3 \triangleright S$  and hence, the result holds by Lemma 5.12.

Hence, we may assume that  $G'$  is isomorphic to an appended 3-star in  $\mathcal{T}_3$ . Thus,  $G$  is an appended star. If  $\Delta \geq 4$ , then we are done by Lemma 5.13. Hence we may assume that  $\Delta = 3$ , implying that  $G$  is an appended 3-star itself. However, for  $G$  not to be isomorphic to any graph in  $\mathcal{T}_\Delta$ , the tree  $G$  must be an appended 3-star of diameter at least 8, in which case, by Lemma 5.16, we are done. ■

By Claim 2, we may assume that  $G'$  is not isomorphic to any tree in  $\mathcal{T}_{\Delta'}$ , for otherwise the desired result follows. Since  $n' \geq 5$  and  $\Delta' \geq 3$ , by the induction hypothesis, we have

$$\gamma^{\text{ID}}(G') \leq \left(\frac{\Delta-1}{\Delta}\right) n'.$$

Thus, by Lemma 5.7, the result holds. □

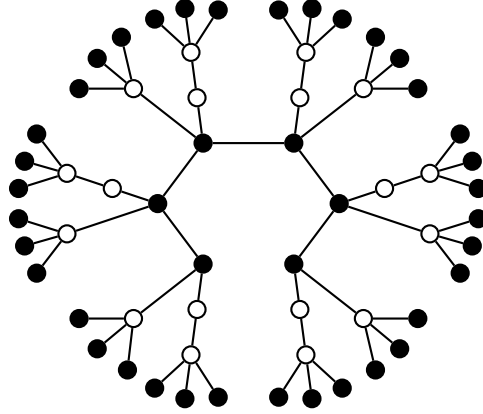


Figure 5.7: Tree  $T_{t,\Delta}$  as in Proposition 5.4 with  $t = 3$  and  $\Delta = 4$ . The set of black vertices constitutes an identifying code of  $T_{t,\Delta}$ .

### 5.2.2.7 Tightness of the bound for trees

In this section, we consider some trees for which Conjecture 5.2 is tight. Clearly, Conjecture 5.2 is tight for every graph in  $\mathcal{T}_\Delta$  (with  $c = 1/\Delta$ ). Moreover, it is tight for some infinite families (with  $c = 0$ ), such as double stars with  $2\Delta - 2$  leaves, where by a *double star* we mean a tree  $G = S_1 \triangleright_u S_2$ , where  $S_1$  is a  $(\Delta - 1)$ -star,  $S_2$  is a  $\Delta$ -star and  $u$  is universal vertex of  $S_1$ . The double star  $G$  has  $n = 2\Delta$  vertices and the smallest identifying code  $C$  of  $G$  has  $2\Delta - 2$  code vertices. Thus,  $|C| = (\frac{\Delta-1}{\Delta})n$ .

Another family of trees for which the conjecture is tight when  $\Delta = 3$  is the 2-corona of a path  $P$  (see [100] or [129, Section 1.3]). We obtain the 2-corona of a graph  $G$  by identifying each vertex  $v$  of  $G$  with a leaf of a 3-vertex path  $P_v$ . We also believe that more examples can be built using the appended stars and paths constructions from Section 5.2.2.3. Moreover, in the regime when both  $n$  and  $\Delta$  are large, we next show that there are trees which almost attain the conjectured bound. Notice that when  $\Delta$  is large, the value  $\frac{\Delta-1+\frac{1}{\Delta-2}}{\Delta+\frac{2}{\Delta-2}}$  is roughly  $\frac{\Delta-1}{\Delta}$ .

**Proposition 5.4.** *Let  $P_t$  be a path of order  $t \geq 3$  and let  $\Delta \geq 4$  be an integer. Let an intermediate tree of order  $n$  be formed by appending onto every vertex of the path  $\Delta - 2 \geq 2$  copies of  $\Delta$ -stars. Thereafter, for each vertex  $p_i$  of the path, subdivide a single edge between  $p_i$  and an adjacent support vertex of the intermediate tree. If  $T_{t,\Delta}$  denotes the resulting tree (see Figure 5.7 for an example with  $t = 6$  and  $\Delta = 4$ ), then*

$$\gamma^{\text{ID}}(T_{t,\Delta}) = \left( \frac{\Delta - 1 + \frac{1}{\Delta-2}}{\Delta + \frac{2}{\Delta-2}} \right) n > \left( \frac{\Delta - 1}{\Delta} \right) n - \frac{n}{\Delta^2}.$$

*Proof.* Let  $T = T_{t,\Delta}$ . For each  $j \in [t]$ , let  $p_j$  be a vertex in the path  $P_t$  and let  $S_i^j$  for  $i \in [\Delta - 2]$  be  $\Delta$ -stars appended onto  $p_j$ . Moreover, let  $u_1^j$  be the universal vertex of  $S_1^j$  and let the edge  $u_1^j p_j$  of the intermediate tree be subdivided by the vertex  $v_1^j$  of  $T$ . We then have  $n = t(\Delta - 2)\Delta + 2 \cdot t$  and the maximum degree of  $T$  is  $\Delta$ . Observe that any identifying code of  $T$  requires at least  $\Delta - 1$  codewords from each star  $S_i^j$ . Furthermore, we need an additional codeword to dominate  $v_1^j$  (if the center of  $S_1^j$  is not chosen yet) or to separate  $v_1^j$  from a leaf of  $S_1^j$  (if the center of  $S_1^j$  is chosen but some leaf of  $S_1^j$  is not). Thus, there are at least  $(\Delta - 2)(\Delta - 1) + 1$  codewords among the vertices in  $V_j(P_t) = \{p_1, v_1^j\} \cup V(S_1^j) \cup V(S_2^j) \cup \dots \cup V(S_{\Delta-2}^j)$ . Moreover, we have  $|V_j(P_t)| = (\Delta - 2)\Delta + 2$ . Thus,

$$\frac{\gamma^{\text{ID}}(T)}{n} \geq \frac{(\Delta - 2)(\Delta - 1) + 1}{(\Delta - 2)\Delta + 2} = \frac{\Delta - 1 + \frac{1}{\Delta-2}}{\Delta + \frac{2}{\Delta-2}}.$$

Moreover, we notice that we may choose as an identifying code of this cardinality, all the leaves and all the path vertices (as illustrated in Figure 5.7 by the set of black vertices). Therefore,

$$\frac{\gamma^{\text{ID}}(T)}{n} = \frac{\Delta - 1 + \frac{1}{\Delta-2}}{\Delta + \frac{2}{\Delta-2}}.$$

We note that

$$\begin{aligned} \frac{\Delta - 1 + \frac{1}{\Delta-2}}{\Delta + \frac{2}{\Delta-2}} &= \frac{1}{\Delta} \left( \frac{\Delta^2 - \Delta + \frac{\Delta}{\Delta-2}}{\Delta + \frac{2}{\Delta-2}} \right) \\ &= \frac{1}{\Delta} \left( \frac{\Delta^3 - 3\Delta^2 + 3\Delta}{\Delta^2 - 2\Delta + 2} \right) \\ &= \frac{\Delta - 1}{\Delta} - \frac{\Delta - 2}{\Delta^3 - 2\Delta^2 + 2\Delta} \\ &> \frac{\Delta - 1}{\Delta} - \frac{\Delta - 2}{\Delta^2(\Delta - 2)} \\ &= \frac{\Delta - 1}{\Delta} - \frac{1}{\Delta^2}. \end{aligned}$$

Hence, the claimed inequality holds.  $\square$

In the previous proposition, we could obtain a slightly better result by adding an additional star for each end of the path. However, that improvement would be of local nature and hence, would improve the result only by a constant while complicating the proof. Note that we may modify the tree  $T_{t,\Delta}$  by adding an edge between the two endpoints of the underlying path  $P_t$ . The resulting graph has the same identification number as the tree  $T_{t,\Delta}$ .

Previously, in [24], an exact value for the identification number of complete  $q$ -ary trees was given. Later in [99], the authors showed that in terms of maximum degree, the complete  $(\Delta - 1)$ -ary tree  $T_\Delta$  has identification number

$$\gamma^{\text{ID}}(T_\Delta) = \left\lceil \left( \frac{\Delta - 2 + \frac{1}{\Delta}}{\Delta - 1 + \frac{1}{\Delta}} \right) n \right\rceil.$$

When we compare this value to the one in Proposition 5.4, we observe that it has slightly smaller multiplier in front of  $n$ .

An observant reader may notice that the root of the tree  $T_\Delta$  has degree of only  $\Delta - 1$ . Hence, we might gain slight improvements by adding some new subtree to it. Let  $r$  be the root of tree  $T_\Delta$  and let  $T'$  be another tree which we join with an edge to  $r$ . Let  $C_\Delta$  (resp.,  $C'$ ) be an identifying code in  $T_\Delta$  (resp.  $T'$ ). Then,  $C = C_\Delta \cup C' \cup \{r\}$  is an identifying code in the combined tree. Furthermore, we may interpret the cardinality of the identifying code also as the *density* of the identifying code in the graph. In particular, we can observe that the density of the code in the combined tree is increased by a meaningful amount if and only if tree  $T'$  is large and a minimum identifying code of  $T'$  has a density larger than that of a minimum identifying code in  $T_\Delta$ . In other words, the tree  $T'$  alone is a better example for a tree with large identifying code than the tree  $T_\Delta$ . Hence, tree  $T_\Delta$  does not help in finding a larger example than we have found in Proposition 5.4.

Another observation we make is that when  $n$  is large and  $\Delta = 4$ , Proposition 5.4 gives  $\gamma^{\text{ID}}(T_{t,4}) = \frac{7}{10}n$ . We do not know any large triangle-free constructions with significant improvements for this case.

### 5.2.3 Triangle-free graphs of given maximum degree

With this section, we continue studying Conjecture 5.2, this time on triangle-free graphs. To state it in a stronger form, we define, for every integer  $\Delta \geq 3$ , a set  $\mathcal{F}_\Delta$  of exceptional graphs of maximum

degree at most  $\Delta$ . For  $\Delta = 3$ , this set contains twelve trees (see Figure 5.2), the cycles on 4 and 7 vertices, and the path on 4 vertices. For every integer  $\Delta > 3$ , it contains exactly the  $\Delta$ -star  $K_{1,\Delta}$ . With that, we state here the theorem that we prove in this section. Notice that a triangle-free graph is necessarily closed-twin-free and hence, is identifiable.

**Theorem 5.8.** *Let  $\Delta \geq 3$  be an integer, and let  $G$  be a connected triangle-free graph of order  $n \geq 3$ . If  $G \in \mathcal{F}_\Delta$ , then*

$$\gamma^{\text{ID}}(G) = \left(\frac{\Delta-1}{\Delta}\right)n + \frac{1}{\Delta}.$$

*On the other hand, if  $G \notin \mathcal{F}_\Delta$  has maximum degree  $\Delta$ , then*

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n.$$

Note that graphs in  $\mathcal{F}_\Delta$  for  $\Delta \geq 4$  have maximum degree  $\Delta$  and the graphs in  $\mathcal{F}_3$  have maximum degree either two or three. In Section 5.2.2, we have shown that Conjecture 5.2 holds for trees (see Theorem 5.6). It turns out that, for triangle-free graphs with maximum degree at least 3, the set of graphs requiring a positive constant  $c$  is exactly the same as the set of trees with maximum degree at least 3 needing a positive constant. Our proof uses the result for trees in Theorem 5.6 as a starting point for an induction. After that, we assume that a triangle-free graph contains at least one cycle containing some edge  $e$ . We remove that edge to construct a graph  $G'$  which, by induction, satisfies the conjecture. Hence,  $G'$  contains a *small* identifying code which we can use to construct another identifying code of the same cardinality for  $G$ . One difficulty for proving the conjecture is the existence of the set of graphs requiring  $c > 0$ . Since  $\mathcal{F}_3$  is the largest among the sets  $\mathcal{F}_\Delta$ , the case  $\Delta = 3$  requires a lot of special argumentation.

### 5.2.3.1 Preliminary results

We use following the lemma multiple times for arguing that some vertices are closed-separated.

**Lemma 5.17.** *Let  $G$  be a triangle-free graph and  $S$ , a subset of vertices of  $G$ . If three vertices  $u, v, w$  inducing a  $P_3$  are in  $S$ , then each of them has a unique closed neighborhood in  $S$ .*

*Proof.* Let  $u, v, w \in S \subseteq V(G)$  induce a  $P_3$  in  $G$  where  $v$  is the middle vertex of the path. Suppose first on the contrary that  $N[u] \cap S = N[x] \cap S$  for some  $x \in V(G) \setminus \{u\}$ . However, now vertices  $x, u$  and  $v$  form a triangle or if  $x = v$ , then  $x, u$  and  $w$  form a triangle. The case with  $w$  is symmetric. Suppose then that  $N[v] \cap S = N[x] \cap S$  for some  $x \in V(G) \setminus \{v\}$ . Now,  $v, w$  and  $x$  form a triangle or if  $x = w$ , then  $u, v$  and  $w$  form the triangle. Hence, the claim follows since  $G$  is triangle-free.  $\square$

We now invoke Lemma 3.3 with the separation type  $A = C$  in the lemma and adapt it only to identifying codes. Moreover, we also adapt the terminologies around Lemma 3.3 the following way to make them specific to identifying codes.

Let  $G$  be a graph and  $X$  and  $Y$  be two (not necessarily disjoint) vertex subsets of  $G$ . Then, any subset  $S$  of  $Y$  induces a partition  $\mathcal{P}(S; X)$  on  $X$  by an equivalence relation  $\sim$  on  $X$  defined by  $u \sim v$  if and only if  $N_G[u] \cap S = N_G[v] \cap S$  for any  $(u, v) \in X \times X$ . If the partition  $\mathcal{P}(S; X)$  is such that each part is a singleton set, that is, for each pair  $u, v \in X$  there exists a vertex  $w \in S$  that separates  $u, v$ , then  $S$  is called an  $(X, Y)$ -*separating set* in  $G$ . Moreover, if such an  $(X, Y)$ -separating set exists, then the set  $X$  is called  $Y$ -*separable*. In addition, we call any set  $S \subseteq Y \subseteq V(G)$  an  $(X, Y)$ -*identifying code* of  $G$  if

- (1)  $S$  is a dominating set of  $X$ ; and
- (2)  $S$  is an  $(X, Y)$ -separating set in  $G$ .



If such an  $(X, Y)$ -identifying code of  $G$  exists, then the set  $X$  is called  $Y$ -*identifiable*. Notice that, if  $X$  is  $Y$ -identifiable and  $S$  is an  $(X, Y)$ -identifying code of  $G$ , then  $Y$  itself is an  $(X, Y)$ -identifying code of  $G$ . Furthermore, the set  $X$  is also  $S$ -identifiable and  $S$  is an  $(X, S)$ -identifying code of  $G$ . In particular, if  $G$  is an identifiable graph and  $S$  is an identifying code of  $G$ , then the vertex set  $V$  is  $V$ -identifiable and  $S$ -identifiable. Moreover, the set  $S$  is a  $(V, V)$ -identifying code and a  $(V, S)$ -identifying code of  $G$ . When  $X$  and  $Y$  are disjoint and induce a bipartite graph, an  $(X, Y)$ -identifying code has been called a *discriminating code* in the literature [57].

With the above terminologies adapted to identifying codes, we now restate Lemma 3.3 in the current context.

**Lemma 5.18.** *Let  $G$  be a graph with vertex subsets  $X$  and  $Y$  such that  $X$  is  $Y$ -identifiable. Then, there is an  $(X, Y)$ -separating set in  $G$  of cardinality at most  $|X| - 1$ . Moreover, there exists an  $(X, Y)$ -identifying code of cardinality at most  $|X|$ .*

*Proof.* If  $|X| = 1$ , then the result follows trivially. Therefore, we assume that  $|X| \geq 2$ . We now inductively construct an  $(X, Y)$ -separating set  $S$  of  $G$  such that  $|S| \leq |X| - 1$ . To begin with, let  $u, v$  be an arbitrary pair of distinct vertices of  $X$  and let  $y \in Y$  such that  $y$  separates the pair: such a vertex  $y$  exists since  $X$  is  $Y$ -identifiable. Then, we let  $S = \{y\}$ . Let  $\mathcal{P}(S; X)$  be the partition induced by  $S$  on  $X$ , where two vertices of  $X$  are in the same part if and only if their closed neighborhood in  $G$  intersects the same subset of  $S$ . Then, for as long as there exists a part  $P$  of  $\mathcal{P}(S; X)$  such that  $u', v' \in P$  for two distinct vertices  $u', v'$  of  $X$ , the construction of  $S$  follows inductively by selecting an element  $y' \in Y$  such that  $y'$  separates  $u', v'$ , and letting  $y' \in S$ . At each step, since  $X$  is  $Y$ -identifiable, such  $y'$  exists. Moreover, we notice that at each inductive step, we have  $|S| \leq |\mathcal{P}(S; X)| - 1$ , since at each step, we increase the number of parts by at least 1, and the cardinality of  $S$  by exactly 1. This implies that we must have  $|S| \leq |X| - 1$ , since  $|\mathcal{P}(S; X)| \leq |X|$ , affirming the first claim.

To prove the second claim, we note that there exists at most one vertex of  $X$  that is not dominated by  $S$ ; for otherwise, if there exist two distinct vertices  $x, x' \in X$  not dominated by  $S$ , it would imply that  $N_G[x] \cap S = N_G[x'] \cap S = \emptyset$  and so,  $S$  would not be an  $(X, Y)$ -separating set of  $X$ , a contradiction. Therefore, let  $x \in X$  be not dominated by  $S$  (if such an  $x$  exists). Since  $X$  is  $Y$ -identifiable, the set  $Y$  dominates every vertex of  $X$ . Therefore, there exists a vertex  $y'' \in N_G[x] \cap Y$ . We let  $y'' \in S$ , thus making  $S$  an  $(X, Y)$ -identifying code of  $G$  with  $|S| \leq |X|$ . This completes the proof.  $\square$

We again recall to the reader here that Lemma 5.18 is a generalization of Bondy's Theorem [29] on induced subsets.

We now define and discuss the exceptional triangle-free graphs of the statement of Theorem 5.8, that is, those in the set  $\mathcal{F}_\Delta$  that require  $c > 0$  in the bound of Conjecture 5.2. The notation for set  $\mathcal{T}_3$  of the collection of 12 trees of maximum degree 3 as in Figure 5.2 defined in Section 5.2.2 will be useful in our proof for the main theorem as we shall build on it to define  $\mathcal{F}_3$ .

**Definition 5.1.** For  $\Delta = 3$ , we define  $\mathcal{T}_3 = \{T_0, T_1, T_2, \dots, T_{11}\}$  to be the collection of 12 trees of maximum degree 3 as in Figure 5.2, and for  $\Delta \geq 4$ , we let  $\mathcal{T}_\Delta = \{K_{1, \Delta}\}$ . Then, for  $\Delta = 3$ , we let  $\mathcal{F}_3 = \mathcal{T}_3 \cup \{P_4, C_4, C_7\}$  and for  $\Delta \geq 4$ , we let  $\mathcal{F}_\Delta = \mathcal{T}_\Delta = \{K_{1, \Delta}\}$ .

For the consistency of this section, we now recall some of the results from Section 5.2.2. First of all, by Corollary 5.1, if  $G$  has maximum degree  $\Delta = 2$ , then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n + \frac{3}{2}$ . We now recall here from Section 5.2.2 our theorem proving Conjecture 5.2 on trees.

**Theorem 5.6.** *Let  $G$  be a tree of order  $n$  and of maximum degree  $\Delta \geq 3$ . If  $G$  is isomorphic to a tree in  $\mathcal{T}_\Delta$ , then, we have*

$$\gamma^{\text{ID}}(G) = \left(\frac{\Delta-1}{\Delta}\right)n + \frac{1}{\Delta}.$$

*On the other hand, if  $G$  is not isomorphic to any tree in the collection  $\mathcal{T}_\Delta$ , then we have*

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n.$$

This implies that for  $\Delta \geq 3$ , we have the following upper bound for graphs in  $\mathcal{F}_\Delta$ .

**Proposition 5.5.** *If  $\Delta \geq 3$  is an integer and  $G$  is a graph of order  $n$  and maximum degree at most  $\Delta$  in  $\mathcal{F}_\Delta$  (possibly, if  $\Delta = 3$ , the maximum degree of  $G$  is 2), then*

$$\gamma^{\text{ID}}(G) = \left( \frac{\Delta - 1}{\Delta} \right) n + \frac{1}{\Delta}.$$

*Proof.* If  $G$  is a tree in  $\mathcal{T}_\Delta$ , then this follows from the first part of Theorem 5.6. Otherwise,  $\Delta = 3$  and  $G \in \{P_4, C_4, C_7\}$ . If  $G \in \{P_4, C_4\}$ , by Corollary 5.1,  $\gamma^{\text{ID}}(G) = 3 = \frac{2}{3}n + \frac{1}{3}$  and if  $G = C_7$ ,  $\gamma^{\text{ID}}(G) = 5 = \frac{2}{3}n + \frac{1}{3}$ .  $\square$

We also recall here Section 5.2.2 the Proposition 5.3 laying out the useful properties regarding the structure of identifying codes of trees in the set  $\mathcal{T}_3$ .

**Proposition 5.3.** *If  $T$  is a tree in  $\mathcal{T}_3$ , then the following properties hold.*

- (i) *If  $T \neq T_2$ , then  $T$  has an optimal identifying code  $C(T)$  containing all vertices of degree at most 2.*
- (ii) *If  $T \notin \{T_2, T_3\}$ , then  $C(T)$  can be chosen as an independent set which contains every vertex of degree at most 2. When we delete any code vertex  $v$  from  $T$ , set  $C(T) \setminus \{v\}$  forms an optimal identifying code of the forest  $T - v$ .*

We shall also need the following property of trees in the family  $\mathcal{T}_3$ .

**Lemma 5.19.** *Let  $T \in \mathcal{T}_3$  be a tree of order  $n$ . If  $e \in E(\overline{T})$  is such that  $T + e$  is triangle-free and  $\Delta(T + e) \leq 3$ , then  $\gamma^{\text{ID}}(T + e) < \frac{2}{3}n$ .*

*Proof.* Assume first that  $T \notin \{T_2, T_3\}$ . Let  $C(T)$  be an optimal identifying code in  $T$  containing every vertex of degree at most 2 which is also an independent set. Such an identifying code exists by Proposition 5.3. Let us denote the end-points of edge  $e$  by  $u$  and  $v$ . Since  $\Delta(T + e) = 3$ , we have  $u, v \in C(T)$ . Observe that we may obtain another identifying code  $C'$  by shifting a single code vertex to any adjacent vertex. This is possible, since every vertex in  $V(T) \setminus C(T)$  is dominated by exactly three vertices in  $C(T)$ . Thus, after shifting, the new code vertex belongs to a  $P_3$ -component in  $T[C']$  and is identified by Lemma 5.17. Furthermore, there now exists exactly one vertex  $w$  in  $V(T) \setminus C(T)$  which is adjacent to one or two vertices in  $C'$ . Since the adjacent vertex in  $C'$  has at least three code vertices in its closed neighborhood, also vertex  $w$  is uniquely identified.

Let us now consider the shifted identifying code  $C'$  so that  $u \notin C'$ . By the previous considerations,  $C'$  is an identifying code in  $T$ . Furthermore, set  $C'$  is an identifying code also in  $T + e$ . Indeed, the only  $I$ -set which is modified by the addition of edge  $e$ , is that of vertex  $u$ . we have  $|N[u] \cap C'| \in \{2, 3\}$ . Note that  $u$  is the only vertex outside of  $C'$  which may have two code vertices in its closed neighborhood. Thus, if  $u$  is not identified, then we have  $|N[u] \cap C'| = 3$ . If  $N[u] \cap C' = N[w] \cap C'$ , then, by Lemma 5.17, we have  $w \notin C'$  (otherwise  $u$  is identified by  $C'$ ). Thus, there are at least two cycles in  $T + e$ , a contradiction.

Assume next that  $T = T_2$ . In this case, the identifying code depicted in Figure 5.2 is also an identifying code in  $T + e$ . Finally, assume that  $T = T_3$ . In this case, the identifying code  $C$  depicted in Figure 5.2 is also an identifying code in  $T + e$ , unless the edge  $e$  is between two leaves at distance 4 in  $T$ . Let us call these two leaves  $u$  and  $v$ . Moreover, let  $u'$  be the leaf at distance 2 from  $u$  and let  $s$  be the support vertex adjacent to  $u$  and  $u'$ . Now,  $\{s\} \cup C \setminus \{u'\}$  is an identifying code in  $T + e$ .  $\square$

We now update the Table 5.1 in Section 5.2.2 to Table 5.2 which lists all triangle-free graphs (known to us) requiring a positive constant  $c$  in Conjecture 5.2.

Graph family	$\Delta$	$c$	Reference
Odd paths	2	$1/\Delta = 1/2$	[24] (Theorem 5.4)
Even paths	2	$2/\Delta = 1$	[24] (Theorem 5.4)
Odd cycles $C_n$ for $n \geq 7$	2	$3/\Delta = 3/2$	[111] (Theorem 5.5)
$C_4$	2	$2/\Delta = 1$	[111] (Theorem 5.5)
$C_5$	2	$1/\Delta = 1/2$	[111] (Theorem 5.5)
$\mathcal{T}_3$	3	$1/\Delta = 1/3$	Theorem 5.6
$K_{1,\Delta}$	$\Delta \geq 3$	$1/\Delta$	Lemma 5.5

Table 5.2: Known triangle-free graphs requiring a positive constant  $c$  for Conjecture 5.2.

### 5.2.3.2 Proof of the main result

In this section, we shall prove our main result, namely Theorem 5.8. Recall its statement.

**Theorem 5.8.** Let  $\Delta \geq 3$  be an integer, and let  $G$  be a connected triangle-free graph of order  $n \geq 3$ . If  $G \in \mathcal{F}_\Delta$ , then

$$\gamma^{\text{ID}}(G) = \left( \frac{\Delta - 1}{\Delta} \right) n + \frac{1}{\Delta}.$$

On the other hand, if  $G \notin \mathcal{F}_\Delta$  has maximum degree  $\Delta$ , then

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n.$$

*Proof.* The first part of the statement holds by Proposition 5.5. For the second part, let  $G$  be a connected triangle-free graph of order  $n$  and size  $m$  with maximum degree  $\Delta \geq 3$  such that  $G \notin \mathcal{F}_\Delta$ . Thus,  $n \geq 5$ . We proceed by induction on  $n + m$  to show that  $\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n$ . Since  $G$  is connected, we note that  $m \geq n - 1$ , and so  $n + m \geq 2n - 1 \geq 9$ . If  $n + m = 9$ , then  $G$  is formed from a star by joining a pendant leaf to another leaf. By Theorem 5.6, we have  $\gamma^{\text{ID}}(G) \leq \frac{2}{3}n$ , in this case. Furthermore, if  $n = m = 5$ , then  $G$  is formed from a 4-cycle joined by a leaf. Observe that such a graph has  $\gamma^{\text{ID}}(G) = 3 < \frac{2}{3}n$ . This establishes the base cases. Let  $n + m \geq 11$ , where  $n \geq 5$ .

For the inductive hypothesis, assume that if  $G'$  is a connected triangle-free graph of order  $n' \geq 3$  and size  $m'$  with  $n' + m' < n + m$  and with maximum degree  $\Delta' = \Delta(G') \geq 3$  such that  $G' \notin \mathcal{F}_{\Delta'}$ , then  $\gamma^{\text{ID}}(G') \leq \left( \frac{\Delta' - 1}{\Delta'} \right) n'$ .

If  $m = n - 1$ , then  $G$  is a tree. Since  $G \notin \mathcal{F}_\Delta$ , by Theorem 5.6, we have  $\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n$ . Hence, we may assume that  $m \geq n$ , for otherwise the desired upper bound follows. Thus, the graph  $G$  contains a cycle edge, that is, an edge that belongs to a cycle in  $G$ . Moreover, since  $\Delta \geq 3$ , the graph  $G$  is not a cycle.

Among all cycle edges in  $G$ , let  $e = uv$  be chosen so that the sum of the degrees of its ends is as large as possible, that is,  $\deg_G(u) + \deg_G(v)$  is maximal. Since  $\Delta(G) \geq 3$ , we have for the edge  $e$  that

$$\deg_G(u) + \deg_G(v) \geq 5.$$

Let

$$G' = G - e.$$

Since  $e$  is a cycle edge of  $G$ , the graph  $G'$  is a connected triangle-free graph of order  $n$ . Let  $\Delta(G') = \Delta'$ , and so  $\Delta' \geq \Delta - 1$ . We note that  $n(G') = n$  and  $m(G') = m - 1$ .

Suppose that  $\Delta' = 2$ . In this case,  $\Delta = 3$  and  $G'$  is either a path or a cycle. Suppose firstly that  $G'$  is a cycle. By the triangle-free condition and our choice of the edge  $e$ , we infer that  $G$  is obtained from a cycle  $C_n$  where  $n \geq 6$  by adding a chord between two non-consecutive vertices on

the cycle in such a way as to create two cycles that both contain the edge  $e$  and both have length at least 4. Assume first that  $n \neq 7$ . By Corollary 5.1(f),  $\gamma^{\text{ID}}(G') \leq \frac{2}{3}n$ . By Observation 5.1,  $\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(G') \leq \frac{2}{3}n$  if  $n$  is odd. Hence, we may consider the case with even  $n \geq 6$ . Let  $G'$  be the cycle  $v_1v_2 \cdots v_nv_1$ . Observe that both the set  $V_{\text{even}}$  of vertices with even subscript and the set  $V_{\text{odd}}$  of vertices with odd subscript are identifying codes of  $G'$  of cardinality  $n/2$ . If the chord in  $G$  is between two vertices with even subscripts, then the set  $V_{\text{odd}}$  is an identifying code in  $G$ , and vice versa. Moreover, if the chord is between a vertex with even subscript and a vertex with odd subscript, then both sets  $V_{\text{even}}$  and  $V_{\text{odd}}$  are identifying codes of  $G$ . Hence,  $\gamma^{\text{ID}}(G) \leq \frac{1}{2}n$ . If  $n = 7$ , then  $\gamma^{\text{ID}}(G) = \gamma^{\text{ID}}(G') - 1 = 4 < \frac{2}{3}n$  by Observation 5.1. Therefore, if  $G'$  is a cycle, then  $\gamma^{\text{ID}}(G) \leq \frac{2}{3}n$ , as desired. Suppose secondly that  $G'$  is a path. In this case,  $n \geq 5$ . For small values of  $n$ , namely  $n \in \{5, 6, 7, 8, 10\}$ , it can readily be checked that  $\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(G') = \lfloor \frac{n}{2} \rfloor + 1 \leq \frac{2}{3}n$ . Hence we may assume that  $n \geq 9$  is odd or  $n \geq 12$  is even. Any identifying code in the path  $G'$  can be extended to an identifying code in  $G = G' + e$  by adding at most one vertex, and so, by Corollary 5.1(a) we have  $\gamma^{\text{ID}}(G) \leq \gamma^{\text{ID}}(G') + 1 = \lfloor \frac{n}{2} \rfloor + 2 \leq \frac{2}{3}n$ . Hence, we may assume that  $\Delta' \geq 3$ , for otherwise the desired bound holds.

Since  $G$  is triangle-free, we note that  $G' \neq K_{1,\Delta'}$  (since otherwise adding back the deleted edge  $e$  would create a triangle in  $G$ ). In particular if  $\Delta' \geq 4$ , then  $G' \notin \mathcal{F}_{\Delta'}$ . If  $\Delta' = 3$  and  $G' \in \mathcal{F}_{\Delta'}$ , then by Observation 5.1 and Lemma 5.19 we infer that  $\gamma^{\text{ID}}(G) < \frac{2}{3}n \leq \left(\frac{\Delta-1}{\Delta}\right)n$ . Hence we may assume that  $G' \notin \mathcal{F}_{\Delta'}$ , for otherwise the desired bound holds. Applying the inductive hypothesis to the graph  $G'$ , we have  $\gamma^{\text{ID}}(G') \leq \left(\frac{\Delta'-1}{\Delta'}\right)n \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .

For notational convenience, let  $N_u = N_G(u) \setminus \{v\}$  and let  $N_v = N_G(v) \setminus \{u\}$ . Since  $G$  is triangle-free and  $uv \in E(G)$ , we note that  $N_u \cap N_v = \emptyset$ . Let  $A$  be the boundary of the set  $\{u, v\}$ , that is,  $A = N_u \cup N_v$  is the set of vertices different from  $u$  and  $v$  that are adjacent to  $u$  or  $v$ . Further, let

$$A_{uv} = A \cup \{u, v\} = N_G[u] \cup N_G[v] \quad \text{and} \quad \bar{A}_{uv} = V(G) \setminus A_{uv}.$$

See Figure 5.8 for an illustration.

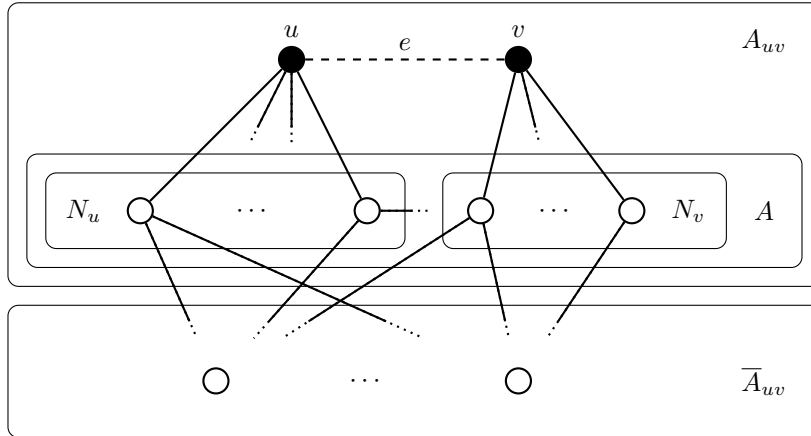


Figure 5.8: The general setting of the proof of Theorem 5.8. Since  $G$  is triangle-free,  $N_u$  and  $N_v$  are independent sets, but there can be edges across them.

Let  $C'$  be an optimal identifying code in  $G'$ , and so  $C'$  is an identifying code in  $G'$  and  $|C'| = \gamma^{\text{ID}}(G')$ . If  $C'$  is an identifying code in  $G$ , then  $\gamma^{\text{ID}}(G) \leq |C'| = \gamma^{\text{ID}}(G') \leq \left(\frac{\Delta-1}{\Delta}\right)n$ . Hence, we may assume that  $C'$  is not an identifying code in  $G$ , for otherwise the desired bound holds.

We will next proceed with proving a series of claims.

■ **Claim 1.** If  $V(G) = A_{uv}$ , then

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n.$$

*Proof of claim.* Suppose that  $V(G) = A_{uv}$ . Recall that by our choice of the edge  $e$ , we have  $\deg_G(u) \geq 2$ ,  $\deg_G(v) \geq 2$ , and  $5 \leq \deg_G(u) + \deg_G(v) \leq 2\Delta$ . In this case since  $V(G) = A_{uv}$ , we have  $n = \deg_G(u) + \deg_G(v) \leq 2\Delta$ . Let  $u'$  be an arbitrary neighbor of  $u$  in  $G$  different from  $v$ , and let  $v'$  be an arbitrary neighbor of  $v$  in  $G$  different from  $u$ . The code  $C = V(G) \setminus \{u', v'\}$  is an identifying code in  $G$ , implying that

$$\begin{aligned} \gamma^{\text{ID}}(G) \leq |C| &= \deg_G(u) + \deg_G(v) - 2 \\ &= \left( \frac{\deg_G(u) + \deg_G(v) - 2}{\deg_G(u) + \deg_G(v)} \right) n \\ &\leq \left( \frac{2\Delta - 2}{2\Delta} \right) n \\ &= \left( \frac{\Delta - 1}{\Delta} \right) n, \end{aligned}$$

yielding the desired upper bound. ■

By Claim 1, we may assume that  $V(G) \neq A_{uv}$ , for otherwise the desired result follows. Let

$$G_{uv} = G - N_G[u] - N_G[v],$$

and so  $V(G_{uv}) = \bar{A}_{uv} = V(G) \setminus (A \cup \{u, v\})$ . Since  $C'$  is an identifying code in  $G'$  but not in  $G$ , a pair of vertices in  $G$  is not identified by the code  $C'$  when adding back the deleted edge  $e$  to the graph  $G'$  to reconstruct  $G$ . Since  $G$  is triangle-free, the only possible pairs of vertices not identified by the code  $C'$  in  $G$  are the pairs  $\{u, v\}$  or  $\{u', v\}$  or  $\{u, v'\}$  where  $u'$  is some neighbor of  $u$  different from  $v$ , and  $v'$  is some neighbor of  $v$  different from  $u$ .

■ **Claim 2.** If for every optimal identifying code  $C'$  in  $G'$  the pair  $\{u, v\}$  is the only pair not identified by the code  $C'$  in  $G$ , then

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n.$$

*Proof of claim.* Let  $C'$  be an optimal identifying code in  $G'$ , and suppose the pair  $\{u, v\}$  is the only pair not identified by the code  $C'$  in  $G$ . Necessarily,  $\{u, v\} \subseteq C'$ , no neighbor of  $u$  different from  $v$  belongs to  $C'$  and no neighbor of  $v$  different from  $u$  belongs to  $C'$ ; that is,  $C' \cap A = \emptyset$ , and so no neighbor of  $u$  in  $G'$  and no neighbor of  $v$  in  $G'$  belongs to  $C'$ . Equivalently,  $C' \setminus \{u, v\} = C' \cap \bar{A}_{uv}$ . See Figure 5.8, where the black vertices belong to  $C'$ , and the other ones do not. We proceed further with a series of subclaims that we will need when proving Claim 2. Since every neighbor of  $u$  (respectively,  $v$ ) is identified by the code  $C'$  in the graph  $G'$ , we infer the following claim.

■■ **Claim 2.1.** Every neighbor of  $u$  (respectively,  $v$ ) in  $G'$  has at least one neighbor that belongs to the set  $C' \setminus \{u, v\}$ .

We shall frequently use the following claim when obtaining structural properties of the graph  $G$ .

■■ **Claim 2.2.** If there exists a vertex  $w \in C' \setminus \{u, v\}$  and a vertex  $z \in A$  such that  $(C' \setminus \{w\}) \cup \{z\}$  is an identifying code in the graph  $G$ , then  $\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n$ .

*Proof of claim.* Consider the set  $(C' \setminus \{w\}) \cup \{z\}$  be as defined in the statement of the claim. If this set is an identifying code in the graph  $G$ , then  $\gamma^{\text{ID}}(G) \leq |C'| \leq \left( \frac{\Delta - 1}{\Delta} \right) n$ . ■■

By Claim 2.2, we may assume that there does not exist a vertex  $w \in C' \setminus \{u, v\}$  and a vertex  $z \in A$  such that  $(C' \setminus \{w\}) \cup \{z\}$  is an identifying code in the graph  $G$ , for otherwise the desired bound holds. Recall that  $C' \cap A = \emptyset$ . In the following claim, we show that there are no  $P_2$ -components in  $G_{uv}$ .

■ ■ **Claim 2.3.** No component in  $G_{uv}$  has order 2.

*Proof of claim.* Suppose that the graph  $G_{uv}$  contains a component  $F$  of order 2, and so  $F$  is isomorphic to  $P_2$ . As observed earlier, in the graph  $G'$  we have  $C' \cap N(u) = C' \cap N(v) = \emptyset$ . In this case, the two vertices in the component  $F$  are not separated by the code  $C'$ , a contradiction. ■ ■

In the following subclaims we consider the case with  $\Delta = 3$  separately. This is due to the more complex structure of set  $\mathcal{F}_3$  compared to sets  $\mathcal{F}_i$  for  $i \geq 4$ .

■ ■ **Claim 2.4.** Let  $\Delta = 3$ . If  $C_7$  is a component in  $G_{uv}$ , then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .

*Proof of claim.* Let  $F = C_7$  be a cycle component in  $G_{uv}$ . Since  $C'$  is an optimal identifying code in  $G'$  and since  $C'$  contains no vertices in the boundary  $A$ , the set  $V(F) \cap C'$  is an identifying code in  $F$ , that is,  $\gamma^{\text{ID}}(F) \leq |V(F) \cap C'|$ . By Theorem 5.5, we have  $\gamma^{\text{ID}}(C_7) = 5$ . Let  $V(F) = \{w_1, w_2, \dots, w_7\}$  and  $E(F) = \{w_7w_1\} \cup \{w_iw_{i+1} \mid 1 \leq i \leq 6\}$ . Observe that each vertex in  $F$  can be adjacent to at most one vertex in  $A$  since  $\Delta = 3$ . Note that, since  $\Delta = 3$ , we have  $1 \leq |N_u| \leq 2$  and  $1 \leq |N_v| \leq 2$ , and so  $|A| \leq 4$ . We denote  $N_u = \{u_1, u_2\}$  and  $N_v = \{v_1, v_2\}$  (if these vertices exist). We further assume that  $|N(F) \cap N_u| \geq |N(F) \cap N_v|$ . Moreover, let  $w_2$  be adjacent to  $u_1 \in N_u$ . Let us consider vertex sets

$$C_1 = (C' \setminus V(F)) \cup \{u_1, w_1, w_4, w_5, w_6\}$$

and

$$C_2 = (C' \setminus V(F)) \cup \{u_1, w_3, w_5, w_6, w_7\}.$$

In the following, we show that at least one of these two sets is an identifying code in  $G$ . Notice that  $|C_1| = |C_2| \leq |C'|$ .

If  $u_1$  is the only vertex of  $A$  adjacent to a vertex in  $F$ , then  $C_1$  is an identifying code in  $G$ . In particular, we may use Lemma 5.17 to see that  $u, v$  and  $u_1$  have unique neighborhoods in  $C_1$ , while  $C_1 \cap V(F)$  forms an identifying code for  $F \setminus \{w_2\}$ , and  $w_2$  is identified by vertices  $u_1$  and  $w_1$ . The remaining vertices are identified by the vertices in  $C'$ .

Assume then that there are two vertices of  $A$  ( $u_1$  and, say,  $x$ ) adjacent to vertices in  $F$ . If  $x$  is in  $N_u$  ( $x = u_2$ ), then  $C_1$  is an identifying code in  $G$ , even if  $x$  is not dominated by a vertex in  $C_1 \cap V(F)$ . Indeed, we have  $u \in N[u_2] \cap C_1$  but  $v, u_1 \notin N[u_2] \cap C_1$ . Thus,  $u_2$  is separated from all other vertices. If  $x \in N_v$ , then  $C_1$  or  $C_2$  is an identifying code in  $G$ . In this case, we choose such a set  $C_i$  ( $i \in \{1, 2\}$ ) so that  $N(x) \cap V(F) \cap C_i \neq \emptyset$ .

Assume next that there are three vertices of  $A$  adjacent to vertices in  $F$ . In this case, due to our assumption that  $|N_u \cap N(F)| \geq |N_v \cap N(F)|$ , we have  $|N_u \cap N(F)| = 2$  and hence, we may assume that  $A \cap N(F) = \{u_1, u_2, v_1\}$ . Again we choose a set  $C_i$  ( $i \in \{1, 2\}$ ), such that  $N(v_1) \cap V(F) \cap C_i \neq \emptyset$ , as our identifying code. Note that as in the previous case, we do not need to dominate  $u_2$  from  $F$ .

Finally, when (all) four vertices of  $A$  are adjacent to vertices in  $F$ , we again choose a set  $C_i$  ( $i \in \{1, 2\}$ ) such that  $N(v_1) \cap V(F) \cap C_i \neq \emptyset$  as our identifying code. As in the previous cases, we do not need to dominate vertices  $u_2$  and  $v_2$  from  $F$ . The argument for  $u_2$  is similar as in the previous cases. Moreover, by Lemma 5.17, vertices  $v$  and  $u$  have unique neighborhoods in  $C_i$ , vertex  $v$  separates  $v_2$  from vertices other than  $u, v$  and  $v_1$ , while  $v_1$  is separated from  $v_2$  by a vertex in  $F$ .

The claim follows in these cases since  $|C_1| = |C_2| \leq |C'|$  and since  $C_1$  or  $C_2$  is an identifying code in  $G$ . ■ ■

■ ■ **Claim 2.5.** Let  $\Delta = 3$ . If  $C_4$  is a component in  $G_{uv}$ , then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .

*Proof of claim.* Let  $F = C_4$  be a cycle component in  $G_{uv}$ . Since  $C'$  is an optimal identifying code in  $G'$  and since  $C'$  contains no vertices in the boundary  $A$ , the set  $V(F) \cap C'$  is an identifying code in  $F$ , that is (by Theorem 5.5),  $\gamma^{\text{ID}}(F) = 3 \leq |V(F) \cap C'|$ . Let  $V(F) = \{w_1, w_2, w_3, w_4\}$  and

$E(F) = \{w_1w_2, w_2w_3, w_3w_4, w_4w_1\}$ . Observe that each vertex in  $F$  can be adjacent to at most one vertex in  $A$  since  $\Delta = 3$ . Recall that by Claim 2.1 every vertex in  $A$  is dominated by two vertices in  $C'$ .

Let us first assume that  $V(F) \subseteq C'$ . Then, we can apply Claim 2.2 to any vertex in  $F$  and its neighbor in  $A$  and the claim follows. Thus, we can now assume that  $|V(F) \cap C'| = 3$ .

Let us first assume that there are one or three vertices in  $A$  adjacent to vertices in  $F$ . Note that each vertex in  $F$  can be adjacent to at most one vertex in  $A$ . If there is one vertex in  $A$  adjacent to a vertex in  $F$ , then we assume that the adjacent vertex in  $F$  is  $w_2 \in F$ . If there are three vertices in  $A$  adjacent to vertices in  $F$ , then we assume that they are adjacent to vertices  $w_1, w_2$  and  $w_3$  in  $F$ . Furthermore, we assume that  $z \in A$  is adjacent to  $w_2$ . We will consider vertex set  $C = (C' \setminus V(F)) \cup \{w_1, w_3, z\}$ . This is an identifying code in  $G$  since all vertices in  $A_{uv}$  are separated from other vertices by  $u$  and  $v$  and vertices  $u, v$  and  $z$  have unique  $I$ -sets with respect to in  $C$  by Lemma 5.17. Moreover, vertices in  $F$  are separated from each other by  $z, w_1$  and  $w_3$  and finally vertices in  $A \setminus \{z\}$  are separated from each other by the same vertices as in  $C'$ .

Assume then that there are four vertices in  $A$  adjacent to vertices in  $F$ . Let us assume without loss of generality that  $\{w_1, w_2, w_3\} \subset C'$  and that  $z \in A$  is adjacent to  $w_2$ . As in the previous case, now  $C = (C' \setminus \{w_2\}) \cup \{z\}$  is an identifying code in  $G$ .

Finally, we have the case where we have exactly two vertices in  $A$  adjacent to vertices in  $F$ . Assume first that these vertices in  $A$  are adjacent to two adjacent vertices of  $F$ , say to  $w_1$  and  $w_2$ . Let  $z \in A$  be a vertex adjacent to  $w_2$ . As in the previous cases, we let  $C = (C' \setminus V(F)) \cup \{w_1, z, w_3\}$ . With the same arguments as in the previous cases,  $C$  is an identifying code in  $G$ .

Let us next consider the case where there are two vertices in  $A$  adjacent to only non-adjacent vertices in  $F$ , say vertices  $w_1$  and  $w_3$ . Observe that now  $w_2$  and  $w_4$  are open twins in  $G'$  and we have exactly two edges between  $A$  and  $F$ . Let  $z \in A$  be a neighbor of  $w_1$ . Assume that  $|N_u| \geq |N_v|$ . Let us denote the neighbors of  $u$  in  $A$  by  $u_1$  and  $u_2$  and neighbors of  $v$  in  $A$  by  $v_1$  and  $v_2$  (if  $v_2$  exists). Notice that if both edges from  $F$  to  $A$  are between  $F$  and  $N_u$  or  $F$  and  $N_v$ , then  $C = \{z, w_2, w_4\} \cup (C' \setminus V(F))$  is an identifying code of  $G$ . Thus, we may assume without loss of generality, that there is an edge from  $w_1$  to  $u_1$  and from  $w_3$  to  $v_1$ . Assume next that for  $u_2$  or  $v_2$  there exists vertex  $c \in C$  such that  $c \in N(v_2) \setminus N(v_1)$  or  $c \in N(u_2) \setminus N(u_1)$ . If the former holds, then  $C_u = \{u_1, w_2, w_4\} \cup (C' \setminus V(F))$  is an identifying code of  $G$  and if the latter holds, then  $C_v = \{v_1, w_2, w_4\} \cup (C' \setminus V(F))$  is an identifying code of  $G$ . Hence, we next assume that  $c$  does not exist. Moreover, if  $v_2$  does not exist, then we may again use identifying code  $C_u$ . Since each vertex in  $A$  is dominated by at least two vertices, there exist vertices  $c_u \in C' \cap N(u_1) \cap N(u_2) \setminus \{u\}$  and  $c_v \in C' \cap N(v_1) \cap N(v_2) \setminus \{v\}$ . Observe that if there exists a vertex  $w_u$  such that  $N[w_u] \cap C' = \{c_u\}$  or a vertex  $w_v$  such that  $N[w_v] \cap C' = \{c_v\}$ , then  $w_u = c_u$  and  $w_v = c_v$ . Indeed, otherwise we would have  $N[w_u] \cap C' = N[c_u] \cap C'$  or  $N[w_v] \cap C' = N[c_v] \cap C'$ . We may also observe that we have either  $\{w_1, w_2, w_3\} \subseteq C'$  or  $\{w_1, w_4, w_3\} \subseteq C'$ . Let us assume, without loss of generality, the former. However, now  $C = \{u_1\} \cup C' \setminus \{u\}$  is an identifying code in  $G$ . Indeed, only  $u, u_1$  and  $u_2$  have lost a code vertex from their neighborhoods compared to  $C'$ . Moreover,  $u$  is the only vertex adjacent to both  $v$  and  $u_1$ ,  $u_1$  is the only vertex adjacent to  $w_1$  and  $c_u$  and finally,  $C \cap N[u_2] = \{c_u\}$  but now  $C \cap N[c_u] \supseteq \{c_u, u_1\}$ . Thus, also  $u_2$  has a unique  $I$ -set. Now, the claim follows. ■■

■■ **Claim 2.6.** Let  $\Delta = 3$ . If  $P_4$  is a component in  $G_{uv}$ , then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .

*Proof of claim.* Let  $F = P_4$  be a path component in  $G_{uv}$ . Since  $C'$  is an optimal identifying code in  $G'$  and since  $C'$  contains no vertices in the boundary  $A$ , the set  $V(F) \cap C'$  is an identifying code in  $F$ , that is, by Theorem 5.4,  $\gamma^{\text{ID}}(F) = 3 \leq |V(F) \cap C'|$ . Let  $V(F) = \{w_1, w_2, w_3, w_4\}$  and  $E(F) = \{w_1w_2, w_2w_3, w_3w_4\}$ .

► **Case 15: there is an edge between  $w_1$  or  $w_4$  and  $A$ .**

Without loss of generality, let it be  $w_1$ . Consider now the graph  $G_1 = G - \{w_2, w_3, w_4\}$ . First

observe that  $G_1$  is connected. Second, if  $G_1 \notin \mathcal{F}_3$ , then by induction we have an identifying code  $C_1 \subseteq V(G_1)$  of cardinality at most  $\frac{2}{3}n(G_1)$ . Moreover, if no neighbor (in  $G$ ) of  $w_2, w_3$  or  $w_4$  is in  $C_1$ , then  $C \cup \{w_2, w_4\}$  is an identifying code of claimed cardinality in  $G$ . If on the other hand  $C_1 \cap (N_G(w_2) \cup N_G(w_3)) \neq \emptyset$ , then at least one of the sets  $C_1 \cup \{w_2, w_4\}$  and  $C_1 \cup \{w_3, w_4\}$  is an identifying code in  $G$ . The case with  $C_1 \cap (N_G(w_4) \cup N(w_3)) \neq \emptyset$  is analogous. Moreover, these identifying codes have the claimed cardinality.

Let us next consider the case with  $G_1 \in \mathcal{F}_3$ . Furthermore, there is a vertex with degree 3 and there are at least six vertices in  $G_1$  since  $\deg(u) + \deg(v) \geq 5$  by our assumption. Thus,  $G_1 \in \mathcal{F}_3 \setminus \{P_4, C_4, C_7, K_{1,3}\} = \mathcal{T}_3 \setminus \{K_{1,3}\}$ . Let us assume first that  $G_1 = T_2$ . In this case, we have  $n = 10$  and we need to find an identifying code containing at most six vertices. Notice that  $\deg(u) + \deg(v) = 5$  and either  $u$  or  $v$  is the support vertex of degree 3 in  $T_2$  and the other one is the adjacent vertex of degree 2. However, since  $w_1$  is a leaf in  $T_2$ , this means that it must have distance of either 1 or 3 to set  $\{u, v\}$ , a contradiction since that distance is actually 2.

Let us next assume that  $G_1 = T_3$ . Now,  $n = 13$  and we need to find an identifying code containing nine vertices. Observe that in this case  $\deg(u) + \deg(v) = 6$ . Since  $w_1$  has distance 2 to set  $\{u, v\}$ , vertex  $w_1$  has to be one of the leaves adjacent to support vertices of degree 2 of  $T_3$ . Moreover, the leaf of  $T_3$  adjacent to the unique support vertex of degree 2 in  $T_3$  as well as that support vertex are not in  $A$  and thus not adjacent to any vertex in  $F$ . However, in this case, to separate that support vertex and its adjacent leaf, the vertex of degree 2 adjacent to the only degree 3 non-support vertex (in  $T_3$ ), must be in any identifying code. However, this vertex is in  $A$ , and this is against our assumption that  $A \cap C' = \emptyset$ , a contradiction.

Assume next that  $G_1 \in \mathcal{T}_3 \setminus \{K_{1,3}, T_2, T_3\}$ . Let  $C_1$  be an optimal identifying code of  $G_1$  such that every vertex with degree at most 2 in  $G_1$  is in  $C_1$  (such a code exists by Proposition 5.3, see Figure 5.2). Since  $G$  is not a tree, there are at least two edges between  $A \cup \{w_1\}$  and  $\{w_2, w_3, w_4\}$ . Hence, there is an edge between  $w_1$  and  $w_2$  and between  $G'_1 = G_1 - \{w_1\}$  and  $\{w_2, w_3, w_4\}$ .

Let us consider the case where there are no edges between  $A$  and  $w_2$  or  $w_3$  but there is an edge to  $w_4$ . In this case, we may consider the identifying code  $C = C_1 \cup \{w_3\}$  which has the claimed cardinality. Notice that this is indeed an identifying code since  $w_1 \in C_1$  and  $w_4$  is dominated by some vertex in  $C_1$ .

Therefore, we may assume from now on that there is an edge between  $A$  and  $w_2$  or  $w_3$ . Let us assume that the edge is from  $A$  to  $w_2$  (the case with an edge to  $w_3$  is similar). Consider now graph  $G''_1 = G - \{w_1, w_3, w_4\}$  together with an optimal identifying code  $C''_1$ . Observe that  $G''_1$  is a tree since  $G_1$  is a tree and  $w_2$  has only one edge to  $A$ . Moreover, we also notice that there are no edges between vertices in  $A$  since  $G_1$  is a tree. Thus, if a 4-cycle contains exactly two vertices in  $F$ , then those vertices are  $w_1$  and  $w_4$ .

If  $G''_1 \notin \mathcal{F}_3$ , then we have two cases based on whether  $w_2 \in C''_1$ . If  $w_2 \in C''_1$ , then  $C''_1 \cup \{w_3, w_4\}$  is an identifying code in  $G$ . Indeed, by Lemma 5.17 vertices  $w_2, w_3$  and  $w_4$  have unique  $I$ -sets. Moreover, also  $w_1$  has a unique neighborhood in  $C''_1 \cup \{w_2\}$  since any vertex of  $A$  adjacent to  $w_2$  is also either in  $C''_1$  or has another neighbor outside of  $A$  in  $C''_1$ .

If  $w_2 \notin C''_1$ , then we use set  $C_{23} = C''_1 \cup \{w_2, w_3\}$ . Since  $C''_1$  is an identifying code in  $G''_1$ , the vertex adjacent to  $w_2$  (call it  $z$ ) in  $A$  is in  $C''_1$  and  $z$  has another code neighbor. Hence, by Lemma 5.17, vertices  $z, w_2$  and  $w_3$  have unique  $I$ -sets. Moreover, since  $w_1$  is adjacent to  $w_2$  and all other vertices in  $N[w_2]$  have unique  $I$ -sets, also  $w_1$  has a unique neighborhood in  $C_{23}$ . Furthermore, also  $w_4$  has a unique neighborhood in  $C_{23}$  since the only other vertex  $x$  adjacent to  $w_3$  which might not be separated from  $w_4$  is in  $A$  and is in  $C''_1$  or has another vertex in  $C''_1$  adjacent to it. Since  $w_3$  and  $w_4$  cannot belong to the same 4-cycle in  $G$  and there are no triangles, vertices  $w_4$  and  $x$  are separated. All the other vertices are pairwise separated by the set  $C''_1$ . Hence,  $C_{23}$  is an identifying code in  $G$ .

Thus, we may assume that  $G''_1 \in \mathcal{F}_3$ . Observe that we may construct  $G''_1$  from  $G_1$  by removing a leaf and then adding a new leaf and vice versa. Hence,  $G_1, G''_1$  are two trees of  $\mathcal{F}_3$  (and thus,  $\mathcal{T}_3$ ) with the same order of at least 6. Observe that for every  $i$  such that  $i \in \{0, 1\}$  or  $i \geq 4$ , there are no



support vertices of degree 2 in  $T_i$  ( $T_i \in \mathcal{T}_3$ ). However, when we remove a leaf from any such tree of  $\mathcal{T}_3$  and add a different leaf to a vertex of degree at most 2, there necessarily exists a support vertex of degree 2 in the resulting tree. Thus, at least one of  $G_1$  and  $G_1''$  must be in  $\{T_2, T_3\}$ . Note that we cannot obtain an isomorphic copy of  $T_3$  by this operation when starting from  $T_3$ , so at most one of  $G_1$  and  $G_1''$  is  $T_3$ . Thus,  $\{G_1, G_1''\} = \{T_1, T_2\}$ ,  $\{G_1, G_1''\} = \{T_2, T_2\}$  or  $\{G_1, G_1''\} = \{T_3, T_4\}$ . Let us first assume that  $G_1, G_1'' = \{T_3, T_4\}$ . Consider the leaf adjacent to the support vertex of degree 2 in  $T_3$ . Notice that the leaf has to belong to  $F$ . Moreover, its distance from set  $\{u, v\}$  is exactly 2. However, now when we form  $T_4$  by removing this leaf and attaching the new leaf, the new leaf is attached adjacent to  $u$  or  $v$  and hence,  $A \cap V(F) \neq \emptyset$ , a contradiction. Thus,  $G_1, G_1'' \in \{T_1, T_2\}$ .

We thus have  $\{G_1, G_1''\} = \{T_1, T_2\}$  or  $\{G_1, G_1''\} = \{T_2, T_2\}$ . In both cases, we have  $\deg(u) + \deg(v) = 5$ . Hence, one of  $u$  or  $v$  is the degree 3 support vertex in  $T_2$  and the other one is the adjacent degree 2 vertex. Moreover, the leaf adjacent to the support vertex of degree 2 in  $T_2$  is in  $F$ . But this vertex should have distance 2 to set  $\{u, v\}$ , a contradiction. This finishes the proof of Case 1.  $\blacktriangleleft$

► **Case 16: there are no edges from  $w_1$  or  $w_4$  to  $A$ .**

Then, there is an edge from  $A$  to  $w_2$  or  $w_3$ ; without loss of generality, assume there is an edge from  $A$  to  $w_2$  and denote  $G_2 = G - \{w_1, w_3, w_4\}$ . Observe that if  $G_2 \notin \mathcal{F}_3$ , then by induction, there exists an identifying code  $C_2$  in  $G_2$  with  $|C_2| \leq \frac{2}{3}n(G_2)$  and  $|C_2| + 2 \leq \frac{2}{3}n$ . Moreover, either  $C_2 \cup \{w_1, w_3\}$  or  $C_2 \cup \{w_2, w_3\}$  is an identifying code in  $G$  depending on whether  $w_2 \in C_2$  or not. Hence, we may assume that  $G_2 \in \mathcal{F}_3$  and in fact  $G_2 \in \mathcal{T}_3$  since  $G_2$  has a vertex of degree 3. If we do not have an edge from  $w_3$  to  $A$ , then  $G$  would be a tree, a contradiction. Hence, we may assume that there exists an edge from  $w_3$  to  $A$ . Let us assume that  $G_2 \neq T_2$  and  $C_2$  contains all vertices of degree at most 2 in  $G_2$ . This is possible by Proposition 5.3. Observe that in this case  $C_2 \cup \{w_3\}$  is an identifying code of claimed cardinality in  $G$ . Hence, we may assume that  $G_2 = T_2$ . Since  $\deg(u) + \deg(v) \geq 5$ , the only leaf of  $T_2$  not in  $A$  is the leaf adjacent to the degree 2 support vertex. However, this vertex has distance 3 to set  $\{u, v\}$ . Thus  $G_2 \neq T_2$ , and the claim follows.  $\blacktriangleleft$

■ ■

■ ■ **Claim 2.7.** Let  $\Delta = 3$ . If  $T \in \mathcal{T}_3$  is a component in  $G_{uv}$ , then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .

*Proof of claim.* Let  $F = T \in \mathcal{T}_3$  be a tree component in  $G_{uv}$ . Since  $C'$  is an optimal identifying code in  $G'$  and since  $C'$  contains no vertices in the boundary  $A$ , the set  $V(F) \cap C'$  is an identifying code in  $F$ , that is,  $\gamma^{\text{ID}}(F) \leq |V(F) \cap C'|$ . Let us call by  $T^2$  the set of vertices of degree at most 2 in  $G[F]$ . Note that only vertices of  $T^2$  can have a neighbor in  $A$ .

Let us first assume that  $F \in \mathcal{T}_3 \setminus \{T_2, T_3\}$ . By Proposition 5.3 we can choose an optimal identifying code  $C_T$  for  $T$  such that  $T^2 \subseteq C_T$  (see Figure 5.2). Moreover, by Proposition 5.3(ii)  $C_T \setminus \{t\}$  is an optimal identifying code in  $T - \{t\}$  for any  $t \in T^2$ . Notice that  $C'_T = (C' \setminus V(F)) \cup C_T$  is an identifying code in  $G'$  and  $|C'| = |C'_T|$ . Notice that Claim 2.2 holds also for  $C'_T$ . Assume first that  $z \in A$  is adjacent to  $t \in T^2$ . If  $C = \{z\} \cup C'_T \setminus \{t\}$  is an identifying code in  $G$ , then the claim follows from Claim 2.2. If  $C$  is not an identifying code, then we will show next that vertex  $t$  must be adjacent to a vertex  $u_1 \in N_u$  and  $v_1 \in N_v$  and  $t$  is the only vertex in  $C'_T$  separating  $u_1$  from another vertex  $u_2 \in N_u$ , and similarly, vertex  $v_2 \in N_v$  from  $v_1 \in N_v$ . Indeed, since  $u, v \in C$ , all vertices of  $A$  are separated from vertices in  $F$ . Moreover, since  $C_T \setminus \{t\}$  is an identifying code in  $F - t$ , set  $C$  is an identifying code in  $F - t$ . Furthermore, vertex  $t$  is the only vertex which has exactly  $z$  in its  $I$ -set. Consequently,  $z, u$  and  $v$  are identified by Lemma 5.17. Therefore, the only vertices which might not be separated belong to  $A$ . Since  $C'_T$  separated all vertices in  $A$ , we require  $t$  to separate some vertices which are not separated by  $z$ . Moreover,  $t$  can have at most two neighbors in  $A$ . If  $t$  is adjacent to only one vertex in  $A$ , then that vertex is  $z$  and  $z \in C$  separates itself from other vertices in  $A$ . Thus,  $t$  is adjacent to two vertices in  $A$ . If both of these vertices are in  $N_u$  (or  $N_v$ ), then  $u$  and  $v$  separate them from other vertices in  $A$ . Moreover,  $z \in C$  separates it itself from other neighbors of  $t$ . Hence,  $t$  is adjacent to a vertex  $u_1 \in N_u$  and  $v_1 \in N_v$ . Since  $u, v \in C$ , the only vertex  $u_1$  that can have the same  $I$ -set with respect to  $C$  is vertex  $u_2 \in N_u$ . A similar statement holds for  $v_1$  and

$v_2 \in N_v$ . Assume first that  $C \setminus \{z\}$  separates  $v_1$  and  $v_2$  in  $G$ . In this case  $(C \cup \{u_1\}) \setminus \{z\}$  is an identifying code in  $G$ . A similar argument holds for  $C \setminus \{z\}$  separating  $u_1$  and  $u_2$  in  $G$ . Hence, we may assume that  $C \setminus \{z\}$  does not separate pairs  $u_1, u_2$  and  $v_1, v_2$ . Thus, we require  $t$  to separate these two pairs in  $C'_T$ , as claimed.

Since  $v_2$  and  $u_2$  are dominated by a vertex in  $C'_T \setminus \{u, v\}$  (Claim 2.1), there is a code vertex  $w_u \in (C'_T \setminus \{u, v\}) \cap N(u_1) \cap N(u_2)$  and a code vertex  $w_v \in (C'_T \setminus \{u, v\}) \cap N(v_1) \cap N(v_2)$ . In this case, we may consider graph  $G'$  and modify the identifying code  $C'_T$  into  $C'' = \{u_1, v_1\} \cup C'_T \setminus \{u, v\}$  which is an identifying code in  $G'$ . This is a contradiction because  $C''$  is an optimal identifying code of  $G'$  such that  $u, v$  are identified by  $C''$  in  $G$ , contradicting the hypothesis of Claim 2.

Assume next that  $F = T_2$ . Denote its unique support vertex of degree 3 by  $s$ . Let the leaves adjacent to  $s$  be denoted by  $l_1$  and  $l_2$  and the third one be  $l_3$ . Finally, let  $f_1, f_2, f_3$  be the three vertices on the path from  $s$  to  $l_3$  (in that order). Assume first that there are no edges between  $A$  and  $f_2, f_3$  or  $l_3$ . Now consider the graph  $G_f = G - \{f_2, f_3, l_3\}$ . Assume first that  $G_f \notin \mathcal{F}_3$  and let  $C_f$  be an optimal identifying code in  $G_f$ . In this case, if  $f_1 \in C_f$ , then  $C_f \cup \{f_2, f_3\}$  is an identifying code of claimed cardinality in  $G$ . If  $f_1 \notin C_f$ , then  $C_f \cup \{f_2, l_3\}$  is an identifying code in  $G$  of claimed cardinality. Assume then that  $G_f \in \mathcal{F}_3$ . Since there is a vertex of degree 3 in  $G_f$ , we have  $G_f \in \mathcal{T}_3$ . However, since there are no edges from  $A$  to  $\{f_2, f_3, l_3\}$ , graph  $G$  is a tree, a contradiction. Hence, we may assume that there is an edge from  $A$  to  $\{f_2, f_3, l_3\}$ .

Let us next consider graph  $G_s = G - \{l_1, l_2, s\}$  and its optimal identifying code by  $C_s$ . Notice that  $G_s$  is connected. Assume first that  $G_s \notin \mathcal{F}_3$ . If one of the three vertices in  $\{l_1, l_2, s\}$  is dominated by a vertex in  $C_s$  in  $G$ , then the set  $C_s$  together with two adjacent vertices from  $\{l_1, l_2, s\}$  such that at least one of them is dominated by a vertex from  $C_s$  is an identifying code of claimed cardinality. If no vertex in  $\{l_1, l_2, s\}$  is dominated by a vertex from  $C_s$ , then we consider  $C_s \cup \{l_1, l_2\}$ . This is an identifying code since  $s$  is separated from all other vertices. Indeed, if  $u \in V(G) \setminus V(F)$  is adjacent to  $l_1$  and  $l_2$ , then it is adjacent also to some vertex in  $C_s \cap (V(G) \setminus V(F))$  while vertex  $s$  cannot be (since it is not dominated by  $C_s$ ). Hence, we may assume that  $G_s \in \mathcal{F}_3$  and more specifically  $G_s \in \mathcal{T}_3$  since there is a vertex of degree 3. Hence, there is a single edge between  $\{f_1, f_2, f_3, l_3\}$  and  $A$ . Furthermore, if there is an edge from  $A$  to  $f_2$  or  $f_3$ , then  $G_s \notin \mathcal{T}_3$ , and if the edge is from  $A$  to  $f_1$  or  $l_3$ , then  $G_s$  has to be  $T_2$ . However, there are at least five vertices in  $A_{uv}$  and hence  $G_s \neq T_2$ . Thus  $G_s \notin \mathcal{F}_3$ .

Let us finally consider the case where  $F = T_3$ . Denote by  $s_1$  and  $s_2$  its two support vertices of degree 3, and by  $f_2$  the support vertex of degree 2. Let leaves  $l_1$  and  $l_2$  be adjacent to  $s_1$ , and leaves  $l_3$  and  $l_4$  be adjacent to  $s_2$ . Furthermore, let the leaf adjacent to  $f_2$  be  $l_5$  and the other vertex adjacent to  $f_2$  be  $f_1$ . Further denote the vertex of degree 3 adjacent to  $f_1$  by  $f_s$ . Assume first that there are no edges from  $A$  to  $\{f_1, f_2, l_5\}$ . Thus,  $G_f = G - \{f_1, f_2, l_5\}$  is a connected graph and let  $C_f$  be an optimal identifying code in  $G_f$ . Notice that if  $G_f \in \mathcal{F}_3$ , then  $G_f \in \mathcal{T}_3$ , and then  $G$  is a tree, a contradiction. Thus,  $G_f \notin \mathcal{F}_3$  and  $C_f$  contains at most two-thirds of the vertices of  $G_f$ . In this case, if  $f_s \in C_f$ , then the set  $C = C_f \cup \{f_1, f_2\}$  is an identifying code in  $G$  and if  $f_s \notin C_f$ , then the set  $C = C_f \cup \{f_1, l_5\}$  is an identifying code in  $G$ . Moreover, both of these sets contain at most two-thirds of the vertices in  $G$ , and we are done. Hence, we may assume from now on that there is an edge from  $A$  to  $\{f_1, f_2, l_5\}$ . Assume then that there are no edges from  $A$  to  $\{s_1, l_1, l_2\}$ . Now, graph  $G_s = G - \{s_1, l_1, l_2\}$  is connected, has a cycle and hence, by induction, also an optimal identifying code  $C_s$  satisfying the two-thirds upper bound. Moreover, set  $C_s \cup \{l_1, l_2\}$  is an identifying code in  $G$ . Hence, there is an edge from  $A$  to  $\{s_1, l_1, l_2\}$ . By symmetry, a similar argument holds for  $\{s_2, l_3, l_4\}$ , so there is an edge from  $A$  to  $\{s_2, l_3, l_4\}$ .

Hence we can assume next that there is an edge from  $A$  to sets  $\{s_1, l_1, l_2\}$ ,  $\{s_2, l_3, l_4\}$  and  $\{l_5, f_1, f_2\}$ . Let us consider graph  $G'_f = G - \{s_2, l_3, l_4\}$  together with an optimal identifying code  $C'_f$  of  $G'_f$ . Notice that  $G'_f$  is connected, has a cycle, and maximum degree 3. Hence,  $G'_f \notin \mathcal{F}_3$  and by induction,  $|C'_f| \leq \frac{2}{3}n(G'_f)$ . Consider set  $C'_f \cup \{l_3, l_4\}$  if no vertex in  $C'_f$  dominates a vertex in  $\{l_3, l_4\}$  and otherwise, set  $C'_f$  together with  $s_2$  and a vertex in  $\{l_3, l_4\}$  that is dominated by  $C'_f$ . Notice that in each case, the corresponding set is an identifying code of claimed cardinality for  $G$ . ■■

By the above claims, we may assume that if  $\Delta = 3$ , then for any component  $F$  in  $G_{uv}$  we have  $\gamma^{\text{ID}}(F) \leq \frac{2}{3}n(F)$ .

■ ■ **Claim 2.8.** Let  $\Delta \geq 4$ . If  $T$  is a  $\Delta$ -star component in  $G_{uv}$ , then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .

*Proof of claim.* Let  $T$  be a  $\Delta$ -star component in  $G_{uv}$ . Let  $V(T) = \{w, w_1, \dots, w_\Delta\}$  and  $w$  be the center vertex of  $T$ . Observe that  $w$  is adjacent only to vertices in  $T$ . Let  $w_1$  be adjacent to a vertex  $z \in A$ . Let  $T' = T - \{w_1\}$  and let  $G_T = G - T'$ . Observe that  $G_T$  is a connected graph and at least one of  $u$  or  $v$  has degree at least 3 in  $G_T$ . Hence, there are at least six vertices in  $G_T$ . Let us assume first that  $G_T \in \mathcal{F}_3$ ; then,  $G_T \in \mathcal{T}_3$ . In this case we have  $\gamma^{\text{ID}}(G_T) \leq \frac{3}{4}n(G_T)$  and  $\gamma^{\text{ID}}(T') \leq \left(\frac{\Delta-1}{\Delta}\right)n(T')$ . Moreover, there are at least three leaves in  $T'$ . Let  $C_T$  be an optimal identifying code in  $G_T$ . Observe that  $C = C_T \cup \{w, w_2, \dots, w_{\Delta-1}\}$  is an identifying code in  $G$  of cardinality at most  $|C| \leq \frac{3}{4}n(G_T) + \left(\frac{\Delta-1}{\Delta}\right)n(T') \leq \left(\frac{\Delta-1}{\Delta}\right)n$ . Furthermore, if  $G_T \notin \mathcal{F}_3$ , since also  $G_T \notin \mathcal{F}_\Delta$ , there exists an identifying code  $C_T$  of  $G_T$  such that  $|C_T| = \gamma^{\text{ID}}(G_T) \leq \left(\frac{\Delta-1}{\Delta}\right)n(G_T)$  and hence, we may again consider  $C = C_T \cup \{w, w_2, \dots, w_{\Delta-1}\}$  as our identifying code for  $G$  and  $|C| \leq \left(\frac{\Delta-1}{\Delta}\right)n$  which completes the proof of the claim. ■ ■

Let us denote the set of isolated vertices in  $G_{uv}$  by  $\mathcal{I}$  and let us further denote  $G_{AI} = G[A_{uv} \cup \mathcal{I}]$ .

■ ■ **Claim 2.9.** Set  $A_{uv}$  is an identifying code in  $G_{AI}$ .

*Proof of claim.* Observe that  $\mathcal{I} \subseteq C'$  (otherwise some vertex of  $\mathcal{I}$  would not be dominated by  $C'$  in  $G'$ ). If set  $A_{uv}$  is not an identifying code in  $G_{AI}$ , then there are open twins  $w_1, w_2$  in  $\mathcal{I}$ . Let  $z \in N(w_1)$ . Notice that  $z \in A$  and hence,  $z \notin C'$ . Since vertices in  $\{w_1, w_2\}$  are open twins, we may consider set  $C = \{z\} \cup C' \setminus \{w_1\}$ . Observe that  $C$  is an identifying code since  $w_2$  separates all the same vertices as vertex  $w_1$ . Hence, we obtain a contradiction from Claim 2.2. Thus, there are no twins in  $\mathcal{I}$  and set  $A_{uv}$  is an identifying code in  $G_{AI}$ . ■ ■

Let us next consider the minimum cardinality of an identifying code in  $G_{AI}$ . Assume that  $|N_u| \geq |N_v|$ . Since  $|N_v| \leq |N_u| \leq \Delta - 1$ , we have  $|A_{uv}| \leq 2\Delta$ . Thus, if  $|A_{uv}| > \left(\frac{\Delta-1}{\Delta}\right)n(G_{AI}) = \left(\frac{\Delta-1}{\Delta}\right)(|A_{uv}| + |\mathcal{I}|)$ , then  $|A_{uv}| > (\Delta - 1)|\mathcal{I}|$ . Hence, this implies that  $|\mathcal{I}| \leq 2$ . By Claim 2.9, the set  $\mathcal{I}$  is  $A$ -identifiable. Hence, by Lemma 5.18, if  $|\mathcal{I}| \leq 2$ , then we require at most two vertices from  $A$  to separate and dominate the vertices in  $\mathcal{I}$ . Moreover, we have  $|A| \geq 3$  due to the assumption  $\deg(u) + \deg(v) \geq 5$ . Notice that set  $A_{uv} \setminus \{z_1, z_2\}$  remains an identifying code of  $G_{AI}$  when  $z_1 \in N_u$  and  $z_2 \in N_v$  and they are not required for dominating or separating vertices in  $\mathcal{I}$ . Indeed, these vertices will be the only ones which are adjacent to exactly  $u$  or  $v$  in the identifying code.

Hence, if  $|\mathcal{I}| = 2$  and  $|A_{uv}| = 2\Delta - 2 - a \leq 2\Delta - 2$  for some  $a \geq 0$ , then  $A_{uv}$  is an identifying code of  $G_{AI}$  that satisfies the conjectured bound for  $G_{AI}$ . Indeed,

$$|A_{uv}| = 2\Delta - 2 - a = \left(\frac{\Delta-1}{\Delta}\right)((2\Delta-2)+2) - a \leq \left(\frac{\Delta-1}{\Delta}\right)n(G_{AI}).$$

If  $|\mathcal{I}| = 2$  and  $|A_{uv}| \geq 2\Delta - 1$ , then we may remove one well-chosen vertex from  $A$  since we require at most two vertices of  $A$  to identify vertices in  $\mathcal{I}$  while  $|A| \geq 3$ , and the resulting set remains an identifying code (as described above) while satisfying the conjectured bound in  $G_{AI}$ . Indeed,  $2\Delta - 2 < \left(\frac{\Delta-1}{\Delta}\right)((2\Delta-2)+3)$  and  $2\Delta - 1 < \left(\frac{\Delta-1}{\Delta}\right)((2\Delta-1)+3)$ . If  $|\mathcal{I}| = 1$  and  $|A_{uv}| \leq 2\Delta - 1$ , then we may remove one well-chosen vertex from  $N_u$ . If  $|\mathcal{I}| = 1$  and  $|A_{uv}| = 2\Delta$ , then we may remove one well-chosen vertex from  $N_u$  and  $N_v$ . Finally, if  $\mathcal{I} = \emptyset$ , then we may just remove any single vertex from both  $N_u$  and  $N_v$  to obtain an identifying code satisfying conjectured bound.

We are now ready to complete the proof of Claim 2. Observe that we have obtained above an identifying code, which contains  $u$  and  $v$ , satisfying the conjectured bound in  $G_{AI}$ . Denote by  $\mathcal{F}_{uv}$  the components in  $G_{uv}$ . Furthermore, observe that if  $F \in \mathcal{F}_{uv}$  has size at least 2 in  $G_{uv}$ , then  $F$  is also a component in  $G - G_{AI}$ . Furthermore, by Claims 2.3, 2.4, 2.5, 2.6, 2.7 and 2.8, every component  $F$  of  $\mathcal{F}_{uv}$  admits an identifying code and we have  $F \notin \mathcal{F}_\Delta$ , and so, by the induction

hypothesis,  $\gamma^{\text{ID}}(F) \leq \left(\frac{\Delta-1}{\Delta}\right) n(F)$ . Let us denote by  $C_F$  an optimal identifying code in component  $F$  and by  $C_{AI}$  an identifying code in  $G_{AI}$  which contains  $u$  and  $v$ , and has cardinality at most  $\left(\frac{\Delta-1}{\Delta}\right) n(G_{AI})$ , as constructed above. Since there are no edges between components of  $\mathcal{F}_{uv}$  and each edge from such a component  $F$  is to a vertex in  $A$ , which are separated from vertices in  $F$  by  $u$  and  $v$ , we have an identifying code

$$C = C_{AI} \cup \bigcup_{F \in \mathcal{F}_{uv}, n(F) \geq 3} C_F$$

and

$$\begin{aligned} |C| &\leq \left(\frac{\Delta-1}{\Delta}\right) n(G_{AI}) + \sum_{F \in \mathcal{F}_{uv}, n(F) \geq 3} \left(\frac{\Delta-1}{\Delta}\right) n(F) \\ &= \left(\frac{\Delta-1}{\Delta}\right) n. \end{aligned}$$

This completes the proof of Claim 2. ■

By Claim 2, we may assume that there exists an optimal identifying code  $C'$  in  $G'$  such that  $\{u', v\}$  or  $\{u, v'\}$  is not identified by  $C'$  in  $G$ , where  $u'$  is some neighbor of  $u$  different from  $v$  and  $v'$  is some neighbor of  $v$  different from  $u$ . Necessarily, in the case where  $\{u', v\}$  is not identified,  $u \in C'$  and  $v \notin C'$ , and if  $X = N_{G'}(v) \cap C'$ , then  $X \neq \emptyset$  (in order for  $C'$  to dominate  $v$  in  $G'$ ) and  $N_{G'}(u') \cap C' = X \cup \{u\}$ . (We have the symmetric facts for the case where  $\{u, v'\}$  is not identified.) Let  $v'$  be an arbitrary vertex in  $X$ , and let  $Q_{uv}$  be the cycle  $uu'v'vu$  in  $G$ . Recall that  $G_{uv} = G - N_G[u] - N_G[v]$ . Further recall that  $N_u = N_G(u) \setminus \{v\}$ ,  $N_v = N_G(v) \setminus \{u\}$ , and  $A = N_u \cup N_v$ . See Figure 5.9 for an illustration.

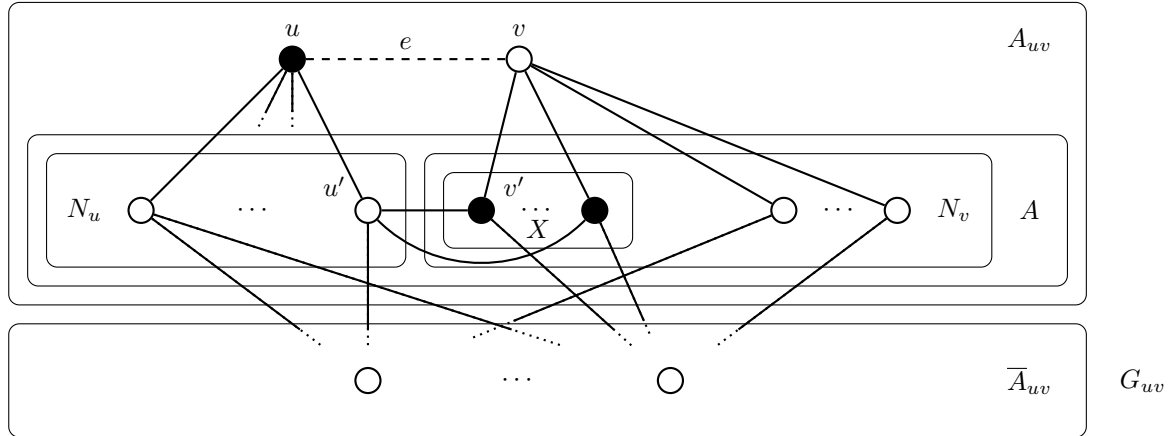


Figure 5.9: The setting of the proof of Theorem 5.8 after applying Claim 2. Here,  $\{u', v\}$  is not identified by  $C'$  (black vertices).

■ **Claim 3.** If  $F$  is a component in  $G_{uv}$  and  $F \in \mathcal{F}_\Delta$ , then

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right) n.$$

*Proof of claim.* Suppose that  $F$  is a component in  $G_{uv}$  and  $F \in \mathcal{F}_\Delta$ . Suppose, firstly, that  $F = K_{1,\Delta}$  where  $\Delta \geq 3$ . Let  $V(F) = \{x, x_1, x_2, \dots, x_\Delta\}$  where  $x$  is the central vertex of  $F$  with leaf neighbors  $x_1, x_2, \dots, x_\Delta$ . Since  $G$  is connected, there is an edge  $f$  joining a vertex  $z \in A$  and a vertex in  $V(F)$ . Renaming vertices if necessary, we may assume that  $f = zx_\Delta$ . Let  $G^* = G - (V(F) \setminus \{x_\Delta\})$ . Let  $G^*$  have order  $n^*$ , and so  $n^* = n - \Delta$ . Further, let  $\Delta(G^*) = \Delta^*$ . Since at least one of  $u$  and  $v$

has degree at least 3 in  $G$ , and since the degrees of  $u$  and  $v$  are the same in  $G$  and in  $G^*$ , we note that  $\Delta^* \geq 3$ . Thus,  $G^*$  is a connected triangle-free graph and  $\Delta \geq \Delta^* \geq 3$ . Moreover,  $G^*$  contains the cycle  $Q_{uv}$  and has order  $n^* \geq 6$ . These properties of  $G^*$  imply that  $G^* \notin \mathcal{F}_{\Delta^*}$ . Applying the inductive hypothesis to  $G^*$ , we have

$$\gamma^{\text{ID}}(G^*) \leq \left( \frac{\Delta^* - 1}{\Delta^*} \right) n^* \leq \left( \frac{\Delta - 1}{\Delta} \right) n^*.$$

Let  $C^*$  be an optimal identifying code in  $G^*$ , and so  $|C^*| = \gamma^{\text{ID}}(G^*)$ . The code  $C^* \cup \{x, x_1, \dots, x_{\Delta-2}\}$  is an identifying code in  $G$  if  $\Delta \geq 4$ , or  $\Delta = 3$  and if  $N_G(x_1) \cap C^* \neq \emptyset$ . Similarly, if  $N_G(x_2) \cap C^* \neq \emptyset$  and  $\Delta = 3$ , then  $C^* \cup \{x, x_2\}$  is an identifying code in  $G$ . Finally, if  $\Delta = 3$  and  $N_G(x_1) \cap C^* = N_G(x_2) \cap C^* = \emptyset$ , then  $C^* \cup \{x_1, x_2\}$  is an identifying code in  $G$ . Hence, we have

$$\begin{aligned} \gamma^{\text{ID}}(G) &\leq |C^*| + \Delta - 1 \\ &= \gamma^{\text{ID}}(G^*) + \Delta - 1 \\ &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* + \left( \frac{\Delta - 1}{\Delta} \right) (n - n^*) \\ &= \left( \frac{\Delta - 1}{\Delta} \right) n, \end{aligned}$$

which yields the desired upper bound in the case when  $F = K_{1,\Delta}$  where  $\Delta \geq 3$ . Hence, we may assume that  $\Delta = 3$  and  $F \in \mathcal{F}_{\Delta} \setminus \{K_{1,3}\}$ . Thus,  $F \in \{P_4, C_4, C_7\} \cup (\mathcal{T}_3 \setminus \{K_{1,3}\})$ .

We next distinguish two cases.

► **Case 1:**  $F \neq P_4$ .

In this case, since the graph  $G$  is connected and  $\Delta = 3$ , one can check that there exists an induced path  $P: x_1x_2x_3$  in  $F$  such that  $G'' = G - V(P)$  is a connected graph. Let  $n'' = n(G'')$ , and so  $n'' = n - 3$ . Let  $\Delta(G'') = \Delta''$ . We note that  $G''$  is a connected triangle-free graph and  $\Delta \geq \Delta'' \geq 3$ . Thus,  $\Delta'' = 3$ . Moreover,  $G''$  contains the cycle  $Q_{uv}$  and has order  $n'' \geq 6$ . In particular,  $G'' \notin \mathcal{F}_3$ . Applying the inductive hypothesis to  $G''$ , we have

$$\gamma^{\text{ID}}(G'') \leq \frac{2}{3}n''.$$

Let  $C''$  be an optimal identifying code in  $G''$ , and so,  $|C''| = \gamma^{\text{ID}}(G'')$ . If  $N_G[x_1] \cap C'' = N_G[x_3] \cap C'' = \emptyset$ , then we consider  $C'' \cup \{x_1, x_3\}$  as a potential identifying code in  $G$ . Note that with this code, each of  $x_1$  and  $x_3$  is only dominated by itself, and no other vertex is in that case. If there exists some vertex  $y$  not separated from  $x_2$  by  $C'' \cup \{x_1, x_3\}$ , then  $y$  must be adjacent to both  $x_1, x_3$ . Since  $G$  is triangle-free,  $y$  is not adjacent to  $x_2$ . Thus,  $y \notin C''$  (otherwise  $y, x_2$  would be separated). Thus,  $y$  is dominated by its third neighbor  $z$ , which is in  $C''$ , and hence,  $z$  is adjacent to  $x_2$ . Since  $\Delta'' = \Delta = 3$ , vertex  $y$  has no other neighbors in  $G$ . Thus,  $y \notin A$ , as in that case,  $y$  should be adjacent to  $u$  or  $v$ , but  $z \notin \{u, v\}$  since  $z$  is adjacent to  $x_2$ . Hence,  $y \in F$  and  $F$  is  $C_4$ . Thus, if  $F \neq C_4$ , we are done. If  $F = C_4$ , we show that  $C'' \cup \{x_1, x_2\}$  is an identifying code in  $G$ . Indeed,  $z, x_1, x_2$  are code vertices inducing a  $P_3$  and we may apply Lemma 5.17 on them. Furthermore,  $x_3$  is separated from all other neighbors of  $x_2$  since they are in the identifying code. Furthermore,  $y$  is adjacent to  $z$  and  $x_1$ . If this is true also for some vertex  $w$ , then  $w \in A$  and  $z$  is adjacent to  $w, y, x_2$  which is not possible since  $\Delta = 3$  and  $z$  is adjacent to  $u$  or  $v$ . Hence,  $C'' \cup \{x_1, x_2\}$  is an identifying code in  $G$ .

Assume next that for an index  $j \in \{1, 3\}$ , say  $j = 1$ , we have  $N_G[x_1] \cap C'' \neq \emptyset$ . Consider the set  $C'' \cup \{x_1, x_2\}$ . If it is an identifying code of  $G$ , then we are done. Thus, assume it is not the case. Vertices  $x_1, x_2$  and the neighbor of  $x_1$  in  $C''$  induce a  $P_3$  and thus by Lemma 5.17, they are uniquely identified. Hence,  $x_3$  is not separated from some other vertex  $w$ . Thus,  $w$  is a neighbor of  $x_2$ , and  $w \notin C''$ . As  $w$  is dominated by  $C''$ , say by  $w'$ , vertices  $w$  and  $x_3$  have  $w' \in C''$  as a second common neighbor. Notice that in this case, vertices  $x_3, x_2, w, w'$  form a 4-cycle and  $N(x_2) = \{x_1, x_3, w'\}$ . However, now set  $C'' \cup \{x_2, x_3\}$  is an identifying code in  $G$ . Indeed,  $x_2, x_3$  and  $w'$  are separated

from other vertices by Lemma 5.17. Moreover, vertex  $w'$  separates  $x_1$  from the two other neighbors of  $x_2$ . Therefore,

$$\begin{aligned}\gamma^{\text{ID}}(G) &\leq |C''| + 2 \\ &= \gamma^{\text{ID}}(G'') + 2 \\ &\leq \frac{2}{3}n'' + \frac{2}{3}(n - n'') \\ &= \frac{2}{3}n,\end{aligned}$$

which yields the desired upper bound.  $\blacktriangleleft$

► **Case 2:**  $F = P_4$ .

Let  $F$  be the path  $x_1x_2x_3x_4$ . If  $x_4$  is adjacent to a vertex in the set  $A$ , then as before, there exists an induced path  $P: x_1x_2x_3$  in  $F$  such that  $G'' = G - V(P)$  is a connected graph of order at least 6 not in  $\mathcal{F}_3$ . Then, the same arguments as in Case 1 apply and we obtain an identifying code of the desired cardinality.

Hence, we may assume that  $x_4$  has degree 1 in  $G$  (with  $x_3$  as its unique neighbor in  $G$ ). Analogously, we may assume that  $x_1$  has degree 1 in  $G$  (with  $x_2$  as its unique neighbor in  $G$ ). Since  $G$  is connected, we may assume, renaming vertices if necessary, that  $x_2$  is adjacent to a vertex in the set  $A$ .

We now consider the graph  $G_F = G - \{x_3, x_4\}$ . Let  $n_F = n(G_F)$ , and so  $n_F = n - 2$ . Let  $\Delta(G_F) = \Delta_F$ . We note that  $G_F$  is a connected triangle-free graph and  $\Delta \geq \Delta_F \geq 3$ . Thus,  $\Delta_F = 3$ . Moreover,  $G_F$  contains the cycle  $Q_{uv}$  and has order  $n_F \geq 7$ . In particular,  $G_F \notin \mathcal{F}_3$ . Applying the inductive hypothesis to  $G_F$ , we have

$$\gamma^{\text{ID}}(G_F) \leq \frac{2}{3}n_F.$$

Let  $C_F$  be an optimal identifying code in  $G_F$ , and so  $|C_F| = \gamma^{\text{ID}}(G_F)$ . If  $x_2 \in C_F$ , then in order to identify the vertices  $x_1$  and  $x_2$ , the code  $C_F$  contains at least one neighbor of  $x_2$  in  $G_F$ . In this case, we let  $C = C_F \cup \{x_3\}$ . If  $x_2 \notin C_F$ , then  $x_1 \in C_F$  and at least one neighbor of  $x_2$  in the set  $A$  belongs to  $C_F$ . In this case, we let  $C = (C_F \setminus \{x_1\}) \cup \{x_2, x_3\}$ . In both cases (using Lemma 5.17 in the second case),  $C$  is an identifying code of  $G$  and  $|C| = |C_F| + 1$ , implying that

$$\begin{aligned}\gamma^{\text{ID}}(G) \leq |C| &= \gamma^{\text{ID}}(G_F) + 1 \\ &< \frac{2}{3}n_F + \frac{2}{3}(n - n_F) \\ &= \frac{2}{3}n.\end{aligned}$$

This completes the proof of Claim 3.  $\blacksquare$

By Claim 3, we will from now on assume that if  $F$  is a component in  $G_{uv}$  of order at least 3, then  $F \notin \mathcal{F}_\Delta$ . Let  $B$  be the set of all vertices that belong to a  $P_1$ -component or to a  $P_2$ -component in  $G_{uv}$ . Furthermore, let us denote  $G^* = G[A \cup B \cup \{u, v\}]$ . Note that it is possible that  $G^* = G$  and hence we do not apply the induction hypothesis directly to  $G^*$ . Recall that  $A = N_u \cup N_v$  and let  $|A| = a$ . Further, let  $|B| = b$ . Thus,  $V(G^*) = A \cup B \cup \{u, v\}$  and  $n^* = a + b + 2$ . Next, in Claim 4, we will show that  $G^*$  admits an identifying code of cardinality at most  $\left(\frac{\Delta-1}{\Delta}\right)n^*$  that contains both vertices  $u$  and  $v$ . Notice that  $G^* \notin \mathcal{F}_\Delta$  since  $G^*$  contains a 4-cycle and a vertex of degree at least 3.

■ **Claim 4.** Graph  $G^*$  admits an identifying code containing vertices  $u$  and  $v$  of cardinality at most

$$\left(\frac{\Delta-1}{\Delta}\right)n^*.$$

*Proof of claim.* Since  $G$  is a connected graph, every vertex in a  $P_1$ -component of  $G[B]$  is adjacent in  $G^*$  to at least one vertex in  $A$ . Moreover, by the connectivity of  $G^*$ , at least one vertex from

every  $P_2$ -component of  $G[B]$  is adjacent in  $G^*$  to at least one vertex in  $A$ . However, it is possible that a  $P_2$ -component of  $G[B]$  contains exactly one vertex that has degree 1 in  $G^*$  (and is therefore not adjacent in  $G^*$  to a vertex from the set  $A$ ). Let  $B_A$  be the set of all vertices in  $B$  that have at least one neighbor in  $A$  in the graph  $G^*$ , and let  $B_L = B \setminus B_A$ . Thus, each vertex in  $B_L$  is a vertex of degree 1 in  $G^*$  that belongs to a  $P_2$ -component of  $G[B]$  and is adjacent to no vertex of  $A$  in  $G^*$ . Possibly,  $B_L = \emptyset$ .

Let  $B = (B_1, \dots, B_k)$  be a partition of the set  $B$  such that the following properties hold.

- (a) All vertices in the set  $B_i \cap B_A$  have the same neighborhood in the set  $A$  for all  $i \in [k]$ , that is, if  $x, y \in B_i \cap B_A$ , then  $N_{G^*}(x) \cap A = N_{G^*}(y) \cap A \neq \emptyset$ .
- (b) Each vertex in the set  $B_i \cap B_L$  has its unique neighbor (in  $G^*$ ) in the set  $B_i \cap B_A$  for all  $i \in [k]$ .
- (c) Vertices in distinct sets  $B_i \cap B_A$  and  $B_j \cap B_A$  have different neighborhoods in  $A$ , that is, if  $x \in B_i \cap B_A$  and  $y \in B_j \cap B_A$  where  $1 \leq i < j \leq k$ , then  $N_{G^*}(x) \cap A \neq N_{G^*}(y) \cap A$ .

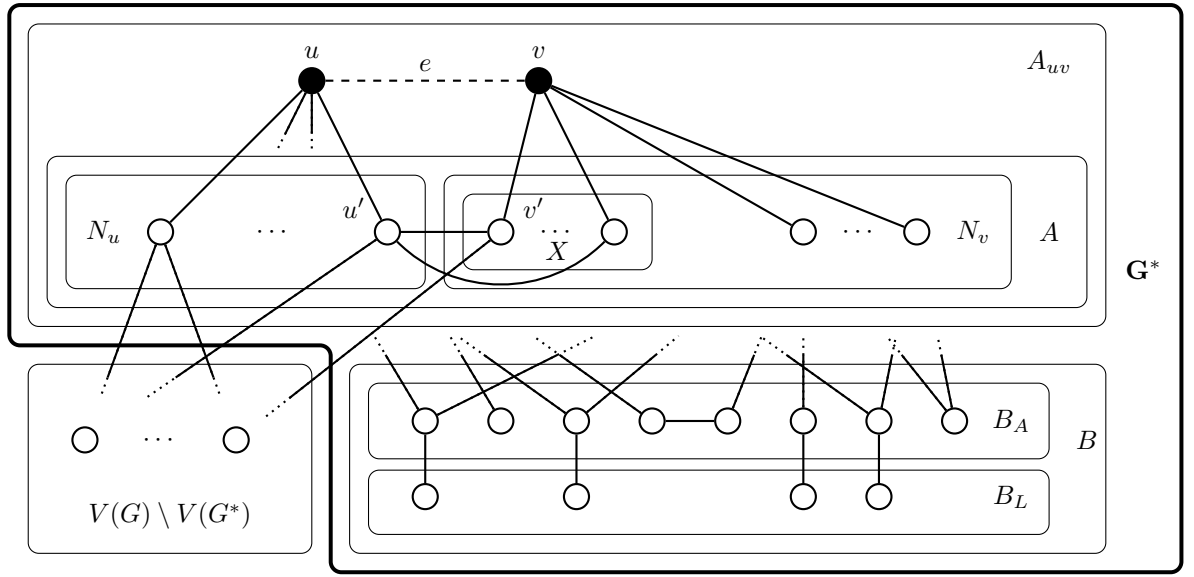


Figure 5.10: The setting of Claim 4 in the proof of Theorem 5.8. The aim is to construct a small identifying code containing both  $u$  and  $v$ .

An illustration is given in Figure 5.10. We note that  $|B_i| = |B_i \cap B_A| + |B_i \cap B_L|$  for all  $i \in [k]$ . Moreover,  $|B_i \cap B_L| \leq |B_i \cap B_A|$  and  $|B_i \cap B_A| \geq 1$  for all  $i \in [k]$ . If  $|B_i \cap B_L| = |B_i \cap B_A|$ , then we observe that the subgraph  $G[B_i]$  induced by the set  $B_i$  consists of vertex-disjoint copies of  $K_2$  (and each such copy of  $K_2$  contains exactly one vertex of degree 1 in  $G$ ). Since each vertex in  $A$  is adjacent to either the vertex  $u$  or  $v$ , each vertex in  $A$  has at most  $\Delta - 1$  neighbors in the set  $B$ , implying that  $|B_i \cap B_A| \leq \Delta - 1$  for all  $i \in [k]$ . Let

$$|B_i| = b_i$$

for all  $i \in [k]$ . Thus, if  $B_i \cap B_L = \emptyset$ , then  $b_i = |B_i \cap B_A| \leq \Delta - 1$ , while if  $B_i \cap B_L \neq \emptyset$ , then  $b_i \leq 2|B_i \cap B_A| \leq 2(\Delta - 1)$  for all  $i \in [k]$ . Recall that

$$b = |B| = \sum_{i=1}^k b_i.$$

Since  $G$  is triangle-free, we note that  $B_i \cap B_A$  is an independent set in  $G^*$  for all  $i \in [k]$ . Thus, if two vertices  $w$  and  $z$  belong to the same  $P_2$ -component in  $G[B]$ , then either  $w \in B_i$  and  $z \in B_j$  where

$1 \leq i, j \leq k$  and  $i \neq j$ , or one of  $w$  and  $z$  belongs to  $B_i \cap B_A$  and the other to  $B_i \cap B_L$  for some  $i \in [k]$ . For  $i \in [k]$ , we now define the set  $B_i^*$  as follows. If  $|B_i \cap B_L| = |B_i \cap B_A|$ , then we define

$$B_i^* = B_i \cap B_L,$$

while if  $|B_i \cap B_L| < |B_i \cap B_A|$ , then we let  $w_i$  be an arbitrary vertex in the set  $B_i \cap B_A$  that is isolated in the subgraph  $G[B_i]$  induced by  $B_i$  (and so,  $w_i$  has no neighbor in  $B_i$ ), and we define

$$B_i^* = (B_i \cap B_L) \cup \{w_i\}.$$

We now define

$$B^* = \bigcup_{i=1}^k B_i^*.$$

Moreover, let  $|B_i^*| = b_i^*$  for all  $i \in [k]$  and

$$b^* = \sum_{i=1}^k b_i^*.$$

An illustration of the set  $B^*$  is given in Figure 5.11.

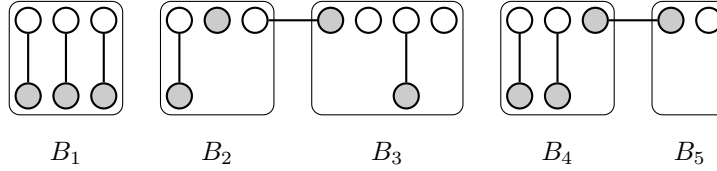


Figure 5.11: The construction of the set  $B^*$  in the proof of Claim 4. The vertices in  $B^*$  are shaded.

■ ■ **Claim 4.1.** The following hold.

- (a)  $b_i^* \geq \frac{b_i}{\Delta-1}$  for all  $i \in [k]$ .
- (b)  $b \leq (\Delta-1)b^*$ .

*Proof of claim.* By our earlier observations, if  $B_i \cap B_L = \emptyset$ , then  $b_i \leq \Delta-1$  where  $i \in [k]$ . Moreover,  $b_i \leq 2(\Delta-1)$  for all  $i \in [k]$ . If  $b_i \leq \Delta-1$  for some  $i \in [k]$ , then

$$\frac{b_i - b_i^*}{b_i} \leq \frac{b_i - 1}{b_i} \leq \frac{\Delta-2}{\Delta-1}.$$

We note that if  $b_i \geq \Delta$  for some  $i \in [k]$ , then  $b_i^* \geq 2$ , and so in this case

$$\frac{b_i - b_i^*}{b_i} \leq \frac{b_i - 2}{b_i} \leq \frac{2(\Delta-1) - 2}{2(\Delta-1)} = \frac{\Delta-2}{\Delta-1}.$$

Thus in both cases,

$$\frac{b_i - b_i^*}{b_i} \leq \frac{\Delta-2}{\Delta-1}.$$

Rearranging terms in the above inequality yields the inequality

$$b_i^* \geq \frac{b_i}{\Delta-1}$$

for all  $i \in [k]$ . This proves property (a) in the statement of the claim. Hence,

$$b = \sum_{i=1}^k b_i \leq (\Delta-1) \sum_{i=1}^k b_i^* = (\Delta-1)b^*,$$



and so property (b) in the statement of the claim holds. ■ ■

Let  $z_i$  be an arbitrary vertex in  $B_i \cap B_A$ , and so  $z_i \in B_i$  and  $z_i$  has a neighbor in the set  $A$  for all  $i \in [k]$ . Let  $Z = \{z_1, z_2, \dots, z_k\}$ . By construction, the set  $A$  identifies the set  $Z$  since every pair of vertices in  $Z$  have distinct neighborhoods in  $A$ . Equivalently, the set  $Z$  is  $(Z, A)$ -identifiable by the set  $A$ . Let  $A^*$  be a subset of  $A$  of minimum cardinality that  $A$ -identifies  $Z$ . By Lemma 5.18,  $1 \leq |A^*| \leq k$ . Since  $b_i^* \geq 1$  for all  $i \in [k]$ , we note that  $b^* \geq k$ , and so  $|A^*| \leq b^*$ . Recall that  $Q_{uv}: uu'v'vu$  is a 4-cycle in  $G$  and hence in  $G^*$ , where  $u'$  is a neighbor of  $u$  in  $G$  different from  $v$ , and  $v'$  is a neighbor of  $v$  in  $G$  different from  $u$ .

**■ ■ Claim 4.2.** If  $A = A^*$ , then there exists an identifying code in  $G^*$  containing vertices  $u$  and  $v$  with cardinality at most

$$\left( \frac{\Delta - 1}{\Delta} \right) n^*.$$

*Proof of claim.* Suppose that  $A = A^*$ . By the minimality of the set  $A^*$ , every vertex in  $A^*$  has a neighbor in  $B$ . Recall that  $V(G^*) = A \cup B \cup \{u, v\}$  and  $n^* = a + b + 2$ . We now let

$$C^* = A \cup \{u, v\} \cup (B \setminus B^*).$$

The set  $C^*$  is an identifying code of  $G^*$ . Indeed,  $C^*$  is a dominating set which is connected in  $G^*$ . Thus, by Lemma 5.17, it separates vertices in  $C^*$ . Furthermore, each vertex in  $B^*$  has a unique neighborhood in  $A \cup (B \setminus B^*)$ .

**■ ■ ■ Claim 4.2.1.** If  $b^* \geq k + 2$ , then  $|C^*| \leq \left( \frac{\Delta - 1}{\Delta} \right) n^*$ .

*Proof of claim.* Suppose that  $b^* \geq k + 2$ . Thus,  $a = |A^*| \leq k \leq b^* - 2$ . By Claim 4.1(b) and our earlier observations, we have

$$\begin{aligned} |C^*| &= a + 2 + b - b^* \\ &= \left( \frac{\Delta - 1}{\Delta} \right) (a + 2 + b) + \frac{1}{\Delta} (a + 2 + b - \Delta b^*) \\ &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* + \frac{1}{\Delta} ((b^* - 2) + 2 + (\Delta - 1)b^* - \Delta b^*) \\ &= \left( \frac{\Delta - 1}{\Delta} \right) n^*, \end{aligned}$$

yielding the desired upper bound. ■ ■ ■

By Claim 4.2.1, we may assume that  $b^* \leq k + 1$ , implying that  $b^* \in \{k, k + 1\}$ . With this assumption,  $b_i^* = 1$  for all  $i \in [k]$ , except for possibly exactly one value of  $i$  satisfying  $b_i^* = 2$ . Recall that every vertex in  $A^*$  has a neighbor in  $B$ . Thus, since  $A = A^*$ , we note in particular that both vertices  $u'$  and  $v'$  have a neighbor in  $B$ . Since  $G^*$  is triangle-free,  $u'$  and  $v'$  do not have a common neighbor. Renaming the partitions in  $B = (B_1, \dots, B_k)$  if necessary, we may assume that  $u'$  is adjacent to a vertex in  $B_1$  and  $v'$  is adjacent to a vertex in  $B_2$ . Since  $u'$  is adjacent to both  $u$  and  $v'$ , it has at most  $\Delta - 2$  additional neighbors, implying that  $|B_1 \cap B_A| \leq \Delta - 2$ . Analogously,  $|B_2 \cap B_A| \leq \Delta - 2$ . Therefore,

$$1 \leq b_1^* \leq b_1 \leq 2(\Delta - 2) \quad \text{and} \quad 1 \leq b_2^* \leq b_2 \leq 2(\Delta - 2).$$

Let

$$B_{3,k} = B \setminus (B_1 \cup B_2) \quad \text{and} \quad B_{3,k}^* = B^* \setminus (B_1^* \cup B_2^*).$$

Further, let  $b_{3,k} = |B_{3,k}|$  and  $b_{3,k}^* = |B_{3,k}^*|$ , and so  $b = b_1 + b_2 + b_{3,k}$  and  $b^* = b_1^* + b_2^* + b_{3,k}^*$ . We note that

$$b_{3,k} = \sum_{i=3}^k b_i \quad \text{and} \quad b_{3,k}^* = \sum_{i=3}^k b_i^* \geq \sum_{i=3}^k 1 = k - 2.$$

We now define a partition  $(V_1, V_2, V_3)$  of  $V(G^*)$  as follows. We let

$$\begin{aligned} V_1 &= B_1 \cup \{u, u'\}, \\ V_2 &= B_2 \cup \{v, v'\}, \\ V_3 &= B_{3,k} \cup (A \setminus \{u', v'\}). \end{aligned}$$

Further we define  $n_i = |V_i|$  for  $i \in [3]$ , and so  $n^* = n_1 + n_2 + n_3$ . We note that  $n_1 = b_1 + 2$ ,  $n_2 = b_2 + 2$ , and  $n_3 = b_{3,k} + a - 2$ . We again consider the identifying code  $C^*$  of  $G^*$  and we let  $C_i^* = C^* \cap V_i$  for  $i \in [3]$ . Thus,

$$|C^*| = \sum_{i=1}^3 |C_i^*|.$$

■■■ **Claim 4.2.2.**  $|C_3^*| \leq \left(\frac{\Delta-1}{\Delta}\right) n_3$ .

*Proof of claim.* By Claim 4.1(a), we have

$$b_{3,k}^* = \sum_{i=3}^k b_i^* \geq \sum_{i=3}^k \frac{b_i}{\Delta-1} = \frac{b_{3,k}}{\Delta-1},$$

and so  $(\Delta-1)b_{3,k}^* \geq b_{3,k}$ , or, equivalently,

$$b_{3,k} - \Delta b_{3,k}^* \leq -b_{3,k}^* \leq -k + 2. \quad (5.1)$$

Recall that  $n_3 = a - 2 + b_{3,k}$  and that  $a \leq k$ . Hence by Inequality (5.1), we have

$$\begin{aligned} |C_3^*| &= |A \setminus \{u', v'\}| + b_{3,k} - b_{3,k}^* \\ &= a - 2 + b_{3,k} - b_{3,k}^* \\ &= \left(\frac{\Delta-1}{\Delta}\right) (a - 2 + b_{3,k}) + \frac{1}{\Delta} (a - 2 + b_{3,k} - \Delta b_{3,k}^*) \\ &\leq \left(\frac{\Delta-1}{\Delta}\right) n_3 + \frac{1}{\Delta} (k - 2 - k + 2) \\ &= \left(\frac{\Delta-1}{\Delta}\right) n_3, \end{aligned}$$

yielding the desired upper bound. ■■■

■■■ **Claim 4.2.3.** If  $b_i^* \geq \frac{b_i}{\Delta-2}$ , then  $|C_i^*| \leq \left(\frac{\Delta-1}{\Delta}\right) n_i$  for  $i \in [2]$ .

*Proof of claim.* Let  $i \in [2]$ . Recall that  $b_i^* \geq 1$  and  $n_i = b_i + 2$ . Suppose that  $b_i^* \geq \frac{b_i}{\Delta-2}$ . Thus,  $(\Delta-2)b_i^* \geq b_i$ , or, equivalently,  $b_i - \Delta b_i^* \leq -2b_i^* \leq -2$  noting that  $b_i^* \geq 1$ . By definition, we have  $C_1^* = \{u, u'\} \cup (B_1 \setminus B_1^*)$  and  $C_2^* = \{v, v'\} \cup (B_2 \setminus B_2^*)$ . Hence,

$$\begin{aligned} |C_i^*| &= 2 + b_i - b_i^* \\ &= \left(\frac{\Delta-1}{\Delta}\right) (2 + b_i) + \frac{1}{\Delta} (2 + b_i - \Delta b_i^*) \\ &\leq \left(\frac{\Delta-1}{\Delta}\right) n_i + \frac{1}{\Delta} (2 - 2) \\ &= \left(\frac{\Delta-1}{\Delta}\right) n_i, \end{aligned}$$

yielding the desired upper bound. ■■■

If  $b_i^* \geq \frac{b_i}{\Delta-2}$  for  $i \in [2]$ , then by Claims 4.2.2 and 4.2.3, we have

$$|C^*| = \sum_{i=1}^3 |C_i^*| \leq \sum_{i=1}^3 \left( \frac{\Delta-1}{\Delta} \right) n_i = \left( \frac{\Delta-1}{\Delta} \right) n^*,$$

yielding the desired upper bound. Hence, we may assume that  $b_i^* < \frac{b_i}{\Delta-2}$  for some  $i \in [2]$ . By symmetry, we may assume that  $b_1^* < \frac{b_1}{\Delta-2}$ . Let

$$|B_1 \cap B_A| = t_1 + t_2 \quad \text{and} \quad |B_1 \cap B_L| = t_2.$$

We note that  $1 \leq t_1 + t_2 \leq \Delta - 2$ .

If  $t_1 \geq 1$  and  $t_2 \geq 1$ , then  $b_1 = t_1 + 2t_2$  and  $b_1^* = 1 + t_2 \geq 2$ , implying that

$$\frac{b_1}{\Delta-2} = \frac{t_2 + (t_1 + t_2)}{\Delta-2} \leq \frac{t_2 + (\Delta-2)}{\Delta-2} = \frac{t_2}{\Delta-2} + 1 \leq \frac{\Delta-3}{\Delta-2} + 1 < 2 \leq b_1^*.$$

If  $t_1 \geq 1$  and  $t_2 = 0$ , then  $b_1 = t_1 \leq \Delta - 2$  and  $b_1^* = 1$ , implying that

$$\frac{b_1}{\Delta-2} \leq \frac{\Delta-2}{\Delta-2} = 1 = b_1^*.$$

If  $t_1 = 0$  and  $t_2 \geq 2$ , then  $b_1 = 2t_2$  and  $b_1^* = t_2 \geq 2$ . Moreover,  $t_2 \leq \Delta - 2$ . Thus,

$$\frac{b_1}{\Delta-2} = \frac{2t_2}{\Delta-2} \leq \frac{2(\Delta-2)}{\Delta-2} = 2 \leq b_1^*.$$

If  $t_1 = 0$ ,  $t_2 = 1$  and  $\Delta \geq 4$ , then  $b_1 = 2$ ,  $b_1^* = 1$  and

$$\frac{b_1}{\Delta-2} = \frac{2}{\Delta-2} \leq \frac{2}{2} = 1 = b_1^*.$$

In all the above four cases, we contradict our assumption that  $b_1^* < \frac{b_1}{\Delta-2}$ . Hence,  $t_1 = 0$ ,  $t_2 = 1$ , and  $\Delta = 3$ . In this case,  $B_1$  induces a  $P_2$ -component in  $G^*$ . Let  $B_1 \cap B_A = \{u_1\}$  and let  $B_1 \cap B_L = \{u_2\}$ , and so  $uu'u_1u_2$  is a path in  $G^*$ . We note that  $u_2$  is a vertex of degree 1 in  $G^*$ . Since  $\Delta = 3$  and  $v'$  is adjacent to  $v$  and  $u'$ , we note that either  $B_2 = P_1$  or  $B_2 = P_2$ . We denote the vertex in  $B_2 \cap N(v')$  by  $v_1$  and if  $v_1$  has another adjacent vertex outside of  $A$ , we denote it by  $v_2$ . By our choice of  $u$  and  $v$  (maximizing  $\deg_G(u) + \deg_G(v)$  for all adjacent  $u, v$  with  $uv$  a cycle edge), we note that both  $u$  and  $v$  have degree 3 in  $G^*$  (otherwise,  $\deg_G(u') + \deg_G(v') > \deg_G(u) + \deg_G(v)$ ). We denote the third neighbor of  $u$  and  $v$  by  $u''$  and  $v''$ , respectively.

Let us assume next that  $|N(u_1) \cap A| = 2$ . Since  $G$  is triangle-free, the other vertex in  $N(u_1) \cap A$  is not  $v'$ . Notice that in this case, we may remove  $u'$  from  $C_1^*$ , resulting in  $C_1^{**}$ , and the set  $C^{**} = C_1^{**} \cup C_2^* \cup C_3^*$  is an identifying code in  $G^*$ . Indeed,  $C^*$  was an identifying code in  $G^*$ . Moreover, every neighbor of  $u'$  is in  $C^{**}$  and vertices in  $C^{**}$  induce a single component in  $G[C^{**}]$ . Hence, every vertex in  $C^{**}$  is separated by Lemma 5.17 while  $u'$  is the unique vertex not in  $C^{**}$  adjacent to vertices  $u$  and  $u_1$ . Notice that  $|C^{**}| \leq \frac{2}{3}n^*$ . Indeed, we have  $n_1 = 4$ ,  $|C_1^{**}| = 2$ ,  $(n_2, |C_2^*|) \in \{(3, 2), (4, 3)\}$  and  $|C_3^*| \leq \frac{2}{3}n_3$  by Claim 4.2.2. Since  $\frac{|C_1^{**}| + |C_2^*|}{n_1 + n_2} \leq \frac{5}{8} < \frac{2}{3}$ , we have  $|C^{**}| \leq \frac{2}{3}n^*$ . Therefore, we may assume that one of  $B_1$  or  $B_2$  is a  $P_2$ -component such that the vertex in  $B_A$  is adjacent to exactly one vertex in  $A$ . We may assume from now on without loss of generality, that  $N(u_1) \cap A = \{u'\}$ .

Recall that in graph  $G'$ , any minimum-ordered identifying code  $C'$  is such that it does not separate either vertices  $\{u, v'\}$  or  $\{v, u'\}$  in  $G$ . Thus,  $C' \cap \{u, v, v', u'\}$  is either  $\{u, v'\}$  or  $\{v, u'\}$ . Furthermore, since  $u_2$  is a leaf and  $u_1$  is adjacent only to  $u'$ , the only vertex which can separate  $u_1$  and  $u_2$  is  $u'$ . Therefore,  $u' \in C'$ . Thus,  $C' \cap \{u, v, v', u'\} = \{u', v\}$  and  $\{u, v'\}$  are not separated in  $G$ . Furthermore, we have  $u'' \notin C'$  (otherwise  $\{u, v'\}$  would be separated by  $C'$  in  $G$ ). Notice that since  $C'$  separates  $u'$  and  $u$ , we have  $u_1, u_2 \in C'$ . However, now the set  $C = \{u\} \cup (C' \setminus \{u_1\})$  is an

identifying code of claimed cardinality in  $G$ . Indeed, vertices  $u', u, v \in C$  are adjacent and are thus separated by Lemma 5.17. Moreover, also vertices  $u_1$  and  $u_2$  have unique  $I$ -sets. This contradicts the properties of  $G$  and completes the proof of Claim 4.2. ■■

Recall that

$$1 \leq |A^*| \leq k \leq b^* \leq b.$$

By Claim 4.2, we may assume that  $A^* \subset A$ , and so  $1 \leq |A^*| < |A| = a \leq 2(\Delta - 1)$ . Let  $\overline{A^*} = A \setminus A^*$ . Thus,  $|\overline{A^*}| = |A| - |A^*| \geq 1$ . We note that either  $|\overline{A^*}| \leq \Delta - 2$  or  $|\overline{A^*}| \geq \Delta - 1$ . Further, either  $\overline{A^*}$  contains a neighbor of  $u$  and a neighbor of  $v$  or  $\overline{A^*} \subset N_{G^*}(u)$  or  $\overline{A^*} \subset N_{G^*}(v)$ . We proceed further with three claims.

■■ **Claim 4.3.** If  $|\overline{A^*}| \leq \Delta - 2$ , then there exists an identifying code in  $G^*$  containing vertices  $u$  and  $v$  with cardinality at most

$$\left(\frac{\Delta - 1}{\Delta}\right)n^*.$$

*Proof of claim.* Suppose that  $|\overline{A^*}| \leq \Delta - 2$ . As observed earlier,  $|A^*| \leq b^*$ . Thus in this case,  $a = |A| = |A^*| + |\overline{A^*}| \leq \Delta + b^* - 2$ . By Claim 4.1(b), we have  $b - (\Delta - 1)b^* \leq 0$ . Let  $x$  be an arbitrary vertex in  $\overline{A^*}$  and let

$$C^* = V(G) \setminus (B^* \cup \{x\}).$$

The code  $C^*$  is an identifying code of  $G^*$ . Indeed,  $C^*$  is a dominating set in  $G^*$ . Furthermore, it is also connected and hence, by Lemma 5.17, separates every vertex in  $C^*$ . Furthermore,  $x$  is the only vertex in  $A$  which does not belong to set  $C^*$  and it is separated by  $\{u, v\}$  from vertices in  $B$ . Finally, vertices in  $B^*$  are separated by  $A^* \cup \{B \setminus B^*\}$ . Next, we consider the cardinality of  $C^*$ :

$$\begin{aligned} |C^*| &= (a + b + 2) - (b^* + 1) \\ &= \left(\frac{\Delta - 1}{\Delta}\right)(a + b + 2) + \frac{1}{\Delta}(a + b - (\Delta - 2) - \Delta b^*) \\ &\leq \left(\frac{\Delta - 1}{\Delta}\right)n^* + \frac{1}{\Delta}((\Delta + b^* - 2) + b - \Delta + 2 - \Delta b^*) \\ &\leq \left(\frac{\Delta - 1}{\Delta}\right)n^* + \frac{1}{\Delta}(b - (\Delta - 1)b^*) \\ &\leq \left(\frac{\Delta - 1}{\Delta}\right)n^*, \end{aligned}$$

yielding the desired upper bound. ■■

■■ **Claim 4.4.** If  $\overline{A^*}$  contains a neighbor of  $u$  and a neighbor of  $v$ , then there exists an identifying code in  $G^*$  containing vertices  $u$  and  $v$  with cardinality at most

$$\left(\frac{\Delta - 1}{\Delta}\right)n^*.$$

*Proof of claim.* Suppose that  $\overline{A^*}$  contains a neighbor,  $u_1$ , of  $u$  and a neighbor,  $v_1$ , of  $v$ . By our earlier observations,  $|\overline{A^*}| = a - |A^*| \leq 2(\Delta - 1) - 1 = 2\Delta - 3$ . Hence,  $a = |A^*| + |\overline{A^*}| \leq b^* + 2\Delta - 3$ . By Claim 4.1(b), we have  $b - (\Delta - 1)b^* \leq 0$ . We now let

$$C^* = V(G^*) \setminus (B^* \cup \{u_1, v_1\}).$$

The set  $C^*$  is an identifying code of  $G^*$ . Indeed,  $C^*$  is a dominating set in  $G^*$ . Furthermore, it is also connected and hence, by Lemma 5.17, separates every vertex in  $C^*$ . Furthermore,  $u_1$  and  $v_1$  are the only vertices in  $A$  which do not belong to set  $C^*$  and are separated by  $\{u, v\}$  from vertices

in  $B$  and from each other. Finally, vertices in  $B^*$  are separated by  $A^* \cup \{B \setminus B^*\}$ . Next, we consider the cardinality of  $C^*$ :

$$\begin{aligned}
 |C^*| &= (a + b + 2) - (b^* + 2) \\
 &= \left( \frac{\Delta - 1}{\Delta} \right) (a + b + 2) + \frac{1}{\Delta} (a + b - 2(\Delta - 1) - \Delta b^*) \\
 &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* + \frac{1}{\Delta} ((b^* + 2\Delta - 3) + b - 2(\Delta - 1) - \Delta b^*) \\
 &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* + \frac{1}{\Delta} (b - (\Delta - 1)b^* - 1) \\
 &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* - \frac{1}{\Delta} \\
 &< \left( \frac{\Delta - 1}{\Delta} \right) n^*,
 \end{aligned}$$

yielding the desired upper bound. ■ ■

By Claim 4.3, we may assume that  $|\overline{A^*}| \geq \Delta - 1$ , for otherwise there exists an identifying code of cardinality at most  $\left( \frac{\Delta - 1}{\Delta} \right) n^*$  in  $G^*$  containing vertices  $u$  and  $v$ .

**■ ■ Claim 4.5.** If  $\overline{A^*} \subseteq N_u$  or  $\overline{A^*} \subseteq N_v$ , then there exists an identifying code in  $G^*$  containing vertices  $u$  and  $v$  with cardinality at most

$$\left( \frac{\Delta - 1}{\Delta} \right) n^*.$$

*Proof of claim.* Suppose that either  $\overline{A^*} \subseteq N_u$  or  $\overline{A^*} \subseteq N_v$ . Renaming  $u$  and  $v$  if necessary, we may assume that  $\overline{A^*} \subseteq N_v$ , and so  $\overline{A^*}$  contains no neighbor of  $u$ . Since  $\overline{A^*} \subseteq N_v$ , we have  $|\overline{A^*}| \leq |N_{G^*}(v)| - 1 \leq \Delta - 1$ . However by our earlier assumption due to Claim 4.3,  $|\overline{A^*}| \geq \Delta - 1$ . Therefore,  $|\overline{A^*}| = \Delta - 1$ , that is,  $\deg_{G^*}(v) = \Delta$  and  $\overline{A^*} = N_{G^*}(v) \setminus \{u\}$ . Thus,

$$a = |A| = |A^*| + |\overline{A^*}| \leq b^* + \Delta - 1.$$

Recall that  $u'v'$  is an edge in  $G^*$ , where  $u'$  is a neighbor of  $u$  in  $G^*$  different from  $v$  and  $v'$  is a neighbor of  $v$  in  $G^*$  different from  $u$ . Since  $u' \in A^*$ , the vertex  $u'$  is adjacent to at least one vertex in  $B$ , and therefore  $u'$  is adjacent to at most  $\Delta - 2$  vertices in  $\overline{A^*}$ . Let  $v''$  be a vertex in  $\overline{A^*}$  that is not adjacent to the vertex  $u'$ . By Claim 4.1(b), we have  $b - (\Delta - 1)b^* \leq 0$ . We now let

$$C^* = V(G^*) \setminus (B^* \cup \{v', v''\}).$$

The set  $C^*$  is an identifying code of  $G^*$ . Indeed,  $C^*$  is a dominating set in  $G^*$ . Furthermore, it is also connected and hence, by Lemma 5.17, separates every vertex in  $C^*$ . Furthermore,  $v'$  and  $v''$  are the only vertices in  $A$  which do not belong to set  $C^*$  and are separated by  $v$  from vertices in  $B$  and by  $u'$  from each other. Finally, vertices in  $B^*$  are separated by  $A^* \cup \{B \setminus B^*\}$ . Next, we consider the cardinality of  $C^*$ :

$$\begin{aligned}
 |C^*| &= (a + b + 2) - (b^* + 2) \\
 &= \left( \frac{\Delta - 1}{\Delta} \right) (a + b + 2) + \frac{1}{\Delta} (a + b - 2(\Delta - 1) - \Delta b^*) \\
 &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* + \frac{1}{\Delta} ((b^* + \Delta - 1) + b - 2(\Delta - 1) - \Delta b^*) \\
 &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* + \frac{1}{\Delta} (b - (\Delta - 1)b^* - (\Delta - 1)) \\
 &\leq \left( \frac{\Delta - 1}{\Delta} \right) n^* - \left( \frac{\Delta - 1}{\Delta} \right) \\
 &< \left( \frac{\Delta - 1}{\Delta} \right) n^*,
 \end{aligned}$$

yielding the desired upper bound. ■ ■

By Claims 4.2, 4.3, 4.4 and 4.5, we have shown that graph  $G^*$  admits an identifying code of cardinality at most  $\left(\frac{\Delta-1}{\Delta}\right)n^*$  containing vertices  $u$  and  $v$ . Hence, Claim 4 follows. ■

If  $G^* = G$ , then the theorem statement follows from Claim 4. If  $G^* \neq G$ , then graph  $G_{uv}$  contains a component of cardinality at least 3. Furthermore, by Claim 3, graph  $G_{uv}$  does not contain any components that belong to the set  $\mathcal{F}_\Delta$ . Hence, we may assume that  $G_{uv}$  contains a component of order at least 3. Next, we finalize the proof by showing that also in this case,  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .

Let us denote by  $\mathcal{K}$  the set of components of order at least 3 in  $G_{uv}$ . By Claim 3,  $K \notin \mathcal{F}_\Delta$  for any  $K \in \mathcal{K}$ . Applying the inductive hypothesis to a component  $K \in \mathcal{K}$  of maximum degree  $\Delta_K$ , component  $K$  satisfies

$$\gamma^{\text{ID}}(K) \leq \left(\frac{\Delta_K-1}{\Delta_K}\right)n(K) \leq \left(\frac{\Delta-1}{\Delta}\right)n(K). \quad (5.2)$$

Let  $C_K$  be an identifying code in  $K$  with minimum cardinality. Observe that  $G^* = G - \bigcup_{K \in \mathcal{K}} V(K)$ . Let  $C^*$  be an identifying code of  $G^*$  with cardinality at most  $\left(\frac{\Delta-1}{\Delta}\right)n^*$  containing vertices  $u$  and  $v$ , that exists by Claim 4.

Let us next consider set  $C = C^* \cup \left(\bigcup_{K \in \mathcal{K}} C_K\right)$ . Notice that we have

$$\gamma^{\text{ID}}(G) \leq |C| = |C^*| + \sum_{K \in \mathcal{K}} |C_K| \leq \left(\frac{\Delta-1}{\Delta}\right)n^* + \sum_{K \in \mathcal{K}} \left(\frac{\Delta-1}{\Delta}\right)n(K) = \left(\frac{\Delta-1}{\Delta}\right)n.$$

Set  $C$  is an identifying code in  $G$  since set  $C^*$  contains vertices  $u$  and  $v$ . Indeed, since  $C$  is a union of multiple identifying codes, each identifying code dominates and pairwise separates the vertices within the corresponding component. Thus, the only problems might be between vertices of different components. In particular, in the case the component  $K$  contains vertex  $z \in C_K$  such that  $N[z] \cap C_K = \{z\}$ , component  $G^*$  contains vertex  $y \in C^*$  such that  $N[y] \cap C^* = \{y\}$ , and  $y$  and  $z$  are adjacent in  $G$ . However, this is not possible since  $y \in A$  and thus,  $N[y] \cap C^* \cap \{u, v\} \neq \emptyset$ .

This completes the proof of Theorem 5.8. □

As an immediate consequence of Theorem 5.8, we have the following corollary.

**Corollary 5.4.** *If  $G$  is a connected, identifiable, triangle-free graph of order  $n$ , then the following holds.*

- (a) *If  $G$  is cubic, then  $\gamma^{\text{ID}}(G) \leq \frac{2}{3}n$ .*
- (b) *If  $G$  is subcubic and  $n \geq 23$ , then  $\gamma^{\text{ID}}(G) \leq \frac{2}{3}n$ .*
- (c) *If  $G$  has maximum degree  $\Delta$  where  $\Delta \geq 4$  is fixed and  $n \geq \Delta + 2$ , then  $\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n$ .*

## 5.2.4 Beyond triangle-free graphs

We now apply Theorem 5.8 to graphs having triangles and obtain a bound weaker than the conjectured one, as follows.

**Corollary 5.5.** *If  $G$  is a connected identifiable graph of order  $n \geq 3$  with maximum degree  $\Delta \geq 3$  such that  $G$  can be made triangle-free by deleting  $t$  edges, then*

$$\gamma^{\text{ID}}(G) \leq \left(\frac{\Delta-1}{\Delta}\right)n + 4t + \frac{1}{\Delta}.$$

*Proof.* We prove the claim by induction on  $t$ . Let us assume that  $E_t \subseteq E(G)$ , with  $|E_t| = t$ , is a smallest set of edges which we can remove from  $G$  so that  $G_t = G - E_t$  is triangle-free. Observe that  $G_t$  is connected and identifiable, since the only connected triangle-free graph that is not identifiable is  $K_2$ , but we assume here that  $n \geq 3$ . By Table 5.1 and Theorem 5.8, the claim holds when  $G$  is triangle-free, that is, for  $t = 0$ . Assume next that the claim holds for every  $t \leq t'$  and let  $t = t' + 1$ . Assume that  $G_t$  has maximum degree  $\Delta_t \leq \Delta$ . We have  $\Delta_t \geq 2$  since  $G_t$  is connected and  $n \geq 3$ .

Let  $C_t$  be an optimal identifying code in  $G_t$ . By Theorem 5.8, if  $\Delta_t \geq 3$ , we have

$$|C_t| \leq \left( \frac{\Delta_t - 1}{\Delta_t} \right) n + \frac{1}{\Delta_t} \leq \left( \frac{\Delta - 1}{\Delta} \right) n + \frac{1}{\Delta}.$$

If  $\Delta_t = 2$ , by Corollary 5.1(e)-(f) and  $n \notin \{4, 7\}$ , then  $|C_t| \leq \frac{2}{3}n \leq \left( \frac{\Delta-1}{\Delta} \right) n$  since  $\Delta \geq 3$ . If  $n = 4$ ,  $|C_t| = 3 \leq \left( \frac{\Delta-1}{\Delta} \right) n + \frac{1}{\Delta}$  and if  $n = 7$ ,  $|C_t| \leq 5 \leq \left( \frac{\Delta-1}{\Delta} \right) n + \frac{1}{\Delta}$  (again since  $\Delta \geq 3$ ). Let edge  $uv \in E_t$  and let us consider graph  $G_t$  together with edge  $uv$ , denoted by  $G'_t$ . Observe that if  $C_t$  is not an identifying code in  $G'_t$ , then the addition of edge  $uv$  modified some code-neighborhoods (this implies that  $u$  or  $v$  is in  $C_t$ ). Therefore, there are at most four vertices which are no longer separated (possibly, vertex  $u$  together with some vertex  $u'$ , and possibly, vertex  $v$  with some vertex  $v'$ ). As we add these edges back one at a time, each time we create at most four new vertices among which some vertex-pairs are not separated by  $C_t$ . Thus, in the graph  $G$ , there is a set  $S$  of at most  $4t$  vertices in which some vertex-pairs are not separated by  $C_t$ . However, since  $G$  is identifiable,  $S$  is  $V(G)$ -identifiable and thus, by Lemma 5.18, we can find an  $(S, V(G))$ -identifying code  $C_S$  of cardinality at most  $|S| \leq 4t$ . The set  $C_t \cup C_S$  is an identifying code of  $G$  of the desired cardinality, proving the claim.  $\square$

## 5.3 Conclusion

The objective of this chapter was to study closed separation in graphs. To that end we study the identifying codes on several graph family by providing bounds on the ID-numbers of these graphs in terms of several parameters like the order of a graph, the number of blocks for a block graph or the maximum degree of a graph. In doing so, we also prove some bounds on ID-numbers of graphs conjectured in the literature in terms of these parameters. We summarize below our work section-wise and also pose some relevant questions.

### 5.3.1 ID-codes of block graphs

It was shown in [10] that ID-CODE can be solved in linear time. In our study on block graphs, we complement this result by presenting tight lower and upper bounds for the ID-codes. We gave bounds both in terms of the order and the number of blocks of a block graph — the latter parameter being equally relevant for block graphs. In particular, we proved Conjecture 5.1 for block graphs posed by Argiroffo et al. in [7]. Moreover, we show that all our bounds are tight by finding block graphs whose ID-numbers attain these bounds. Also, noting that block graphs are a subfamily of chordal graphs, it is natural to ask the following question.

**Open Problem 5.1.** *What are the tight upper and lower bounds on the ID-numbers of ID-admissible chordal graphs?*

### 5.3.2 ID-codes of trees of given maximum degree

In this chapter, we have made significant progress towards Conjecture 5.2 by proving it for all trees. Moreover, we have precisely characterized the trees that need  $c > 0$  in the conjectured bound. It is interesting that for every fixed  $\Delta \geq 3$ , there is only a finite list of such trees. We close with the following list of open questions and problems.

**Question 5.1.** For every fixed  $\Delta \geq 3$ , if  $G$  is a connected identifiable graph of order  $n$  and of maximum degree  $\Delta$ , then is it true that

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n,$$

except for a finite family of graphs?

**Open Problem 5.2.** Characterize the trees  $T$  of order  $n$  with maximum degree  $\Delta$  satisfying

$$\gamma^{\text{ID}}(T) = \left( \frac{\Delta - 1}{\Delta} \right) n.$$

**Open Problem 5.3.** Characterize the trees  $T$  of order  $n \geq 3$  satisfying  $\gamma^{\text{ID}}(T) = n - \gamma(T)$ , that is, characterize the trees  $T$  that achieve equality on the upper bound in Theorem 5.7.

**Open Problem 5.4.** Determine families of graphs  $G$  of order  $n \geq 3$  satisfying  $\gamma^{\text{ID}}(T) \leq n - \gamma(T)$ , that is, does Theorem 5.7 hold for a larger family of graphs (for example some subfamily of bipartite graphs containing all trees)?

As mentioned before, in Section 5.2.3, we use our result for trees to prove Conjecture 5.2 for all triangle-free graphs (with the same list of graphs requiring  $c > 0$  when  $\Delta \geq 3$ ).

### 5.3.3 ID-codes of triangle-free graphs of given maximum degree

In section 5.2.3 dealing with triangle-free graphs, we managed to prove Conjecture 5.2 for all graphs of this graph class. In fact, we proved (see Theorem 5.8) the following stronger result. Let  $G$  be a connected, identifiable, triangle-free graph of order  $n \geq 3$  with maximum degree  $\Delta$ . If  $\Delta \geq 4$ , then we proved that  $\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n$ , except for one exceptional graph, namely the star  $K_{1,\Delta}$ . Moreover, if  $\Delta \leq 3$ , then  $\gamma^{\text{ID}}(G) \leq \frac{2}{3}n$ , unless  $G$  belongs to a forbidden family that contains fifteen graphs (all of order at most 22):  $P_4$ ,  $C_4$ ,  $C_7$  and the twelve trees of maximum degree 3 from  $\mathcal{T}_3$ .

In the special case when  $G$  is a triangle-free cubic graph, this implies that  $\gamma^{\text{ID}}(G) \leq \frac{2}{3}n$  always holds. This establishes a best possible upper bound for triangle-free cubic graphs, since  $\gamma^{\text{ID}}(K_{3,3}) = 4$  (we do not know other cubic graphs for which this bound is tight exist). The previously best known upper bounds (prior to this work) when  $G$  is triangle-free subcubic and cubic were,  $\gamma^{\text{ID}}(G) \leq \frac{8}{9}n$  and  $\gamma^{\text{ID}}(G) \leq \frac{5}{6}n$ , respectively (see [87, Corollary 4.46]; the proof used the technique developed in [104]). Towards a positive resolution of Conjecture 5.2, it would be interesting to prove it for all cubic graphs:

**Open Problem 5.5.** Is Conjecture 5.2 true for all connected and identifiable cubic graphs?

When it comes to general graphs, the list of exceptional graphs is larger than  $\mathcal{F}_\Delta$ . Indeed, to mention some graphs containing triangles, the complements of half-graphs and related constructions defined in [90] do require  $c > 0$ . Nevertheless, those constructions have maximum degree  $\Delta$  very close to the number  $n$  of vertices ( $n - 1$  or  $n - 2$ ), and, for any given  $\Delta$ , there is only a finite number of such examples. Thus, it is possible that even for the general case, if the conjecture is true, the list of graphs requiring  $c > 0$  is also finite for every fixed value of  $\Delta$ .

The positive constant  $c$  in Conjecture 5.2 that the half-graphs and their related constructions described in [90] require are also bounded above by  $\frac{2}{\Delta}$ . These half graphs and the triangle-free graphs mentioned in Table 5.2 are all the graphs that are known to us that require a positive constant in Conjecture 5.2. In other words, all graphs known to us that require  $c > 0$  have  $c \leq 3/2$  (which is reached only by odd cycles), and when  $\Delta \geq 3$ , in fact  $c \leq 1/3$ . Are there graphs that require higher values of  $c$ ? Another way to formulate these constants is in terms of the maximum degree.



**Question 5.2.** *Does there exist graphs that require the constant  $c$  to be larger than  $3/\Delta$ ?*

By our results, such graphs requiring the constant  $c$  to be greater than  $3/\Delta$  would necessarily contain triangles. Note that it seems necessary to understand those graphs needing  $c > 0$ , in order to prove the conjecture.

## Chapter 6

# Open separation in graphs

In this chapter, we look at open separation as a separation property and study open-separating dominating codes and open-separating total-dominating codes in graphs. We first look at the problem of OTD-codes in graphs in Section 6.1 since, historically, it appears first in the literature of identification problems. Next, in Section 6.2, we look at OD-codes.

To begin with, using the various definitions related to open-separating sets, we recall here (all in one place) the equivalent conditions to imply when a vertex subset of an open-separable graph  $G$  is an open-separating set of  $G$ .

**Remark 6.1.** *Let  $G$  be an open-separable graph and let  $S$  be a vertex subset of  $G$ . Then, the following assertions are equivalent.*

- (1)  $S$  is an open-separating set of  $G$ .
- (2) Each vertex in  $V(G)$  has a unique neighborhood in  $S$ .
- (3) For all distinct  $u, v \in V(G)$ , we have  $N_G(u) \cap S \neq N_G(v) \cap S$ .
- (4)  $S$  has non-empty intersection with  $\Delta_O(G; u, v) = N_G(u) \Delta N_G(v)$  for all distinct  $u, v \in V(G)$ .

## 6.1 Open separation with total domination

In this section, we look at open-separating total-dominating codes of graphs in some special graph families. In Section 6.1.1, we consider OTD-codes of block graphs and study both upper and lower bounds on the OTD-numbers of graphs of this class. In Section 6.1.2, we provide the exact values of the OTD-numbers of  $n$ -cycles (with  $n \neq 4$ ). In Section 6.1.3, we study OTD-codes of  $C_4$ -free graphs in relation with their orders, say  $n$  and their maximum degree, say  $\Delta$ , and provide upper bounds on their OTD-numbers in terms of both  $n$  and  $\Delta$ . All bounds we study in this section are tight with the OTD-numbers of arbitrarily large-ordered graphs attaining these bounds.

The following remark shows that in order to check if a total-dominating  $S$  of a graph  $G$  is an OTD-code of  $G$ , we do not need to check if  $S$  open-separates every pair of distinct vertices of  $G$  but only those which are at a distance of at most 2 between them. The following remark was also shown in [11].

**Remark 6.2.** *Let  $G$  be an OTD-admissible graph. A total-dominating set  $C$  of  $G$  is an OTD-code of  $G$  if and only if  $C$  open-separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \leq 2$ .*

*Proof.* The necessary condition for the statement follows immediately from the definition of an OTD-code. We, therefore, prove the sufficient condition. Thus, it is enough to show that  $C$  open-separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . So, assume  $u$  and  $v$  to be

a pair of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . Since  $C$  is a total-dominating set of  $G$ , the vertex  $u$  has a neighbor, say  $w$ , in  $C$ . However,  $w$  is not a neighbor of  $v$  (since  $d_G(u, v) \geq 3$ ) and hence,  $w$  must be a separating  $C$ -codeword of the pair  $u, v$  in  $G$ . This proves the result.  $\square$

## 6.1.1 Block graphs

In this section, we first prove a tight upper bound for the OTD-numbers of block graphs. We then also prove two tight lower bounds on OTD-numbers of block graphs, one, in terms of the number of vertices of the graph and, two, in terms of the number of blocks of the graph. We refer the reader to Section 2.1.3 for the basic definitions and notations concerning block graphs.

The results of this section have also appeared in [47].

### 6.1.1.1 Upper bounds on OTD-numbers of block graphs

In this section, we focus our attention on upper bounds for OTD-numbers of block graphs. Before we get to our results, we recall the 4-path  $P_4$  and define the bull graph.

1. The *4-path* (or  $P_4$  in symbol) is a graph defined by its vertex set  $V(P_4) = \{p_1, p_2, p_3, p_4\}$  and its edge set  $E(P_4) = \{p_1p_2, p_2p_3, p_3p_4\}$ .
2. The *bull graph* (or  $Q_5$  in symbol) is a graph defined by its vertex set  $V(Q_5) = \{b_1, b_2, b_3, b_4, b_5\}$  and its edge set  $E(Q_5) = \{b_1b_2, b_2b_3, b_3b_4, b_4b_5, b_2b_4\}$ . See Figure 6.1 for a depiction of a bull graph.

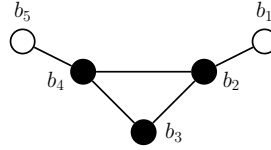


Figure 6.1: The Bull graph  $Q_5$ . The set of black vertices constitute an OTD-code of  $Q_5$ .

We note here that both  $P_4$  and  $Q_5$  are block graphs with articulation vertices  $p_2$  and  $p_3$  for  $P_4$  and  $b_2$  and  $b_4$  for  $Q_5$ . For  $P_4$ , the vertices  $p_1$  and  $p_4$  are called the *leaf vertices*; and for  $Q_5$ , the vertices  $b_1$  and  $b_5$  are the leaf vertices. Assume  $G'$  to be any graph and  $X$  to be a graph which is either a copy of  $P_4$  or  $Q_5$ . For a fixed vertex  $q \in V(G')$ , we define a new graph  $G' \triangleright_q X$  to be the graph obtained by identifying the vertex  $q$  with an articulation vertex of  $X$  (see Figures 6.2a and 6.2b for examples of  $K_4 \triangleright_q P_4$  and  $K_4 \triangleright_q Q_5$ , respectively). As a matter of reference, we call the new vertex in  $G' \triangleright_q X$  — obtained as a result of identifying two vertices — as the *quotient vertex*; and continue to refer to the quotient vertex as  $q$  itself.

We now turn to our results. Firstly, it is easy to establish the following.

**Lemma 6.1.** *If  $P$  is a 4-path, then  $\gamma^{\text{OTD}}(P) = 4$ .*

**Lemma 6.2.** *If  $B$  is a bull graph, then  $\gamma^{\text{OTD}}(B) = 3$ .*

*Proof.* Let  $V(B) = \{b_1, b_2, b_3, b_4, b_5\}$ , where  $b_2$  and  $b_4$  are the articulation vertices; and  $b_1$  and  $b_5$  are the leaf vertices of  $B$ . Then it is easy to check that  $\{b_2, b_3, b_4\}$  is an OTD-code of  $B$  and hence  $\gamma^{\text{OTD}}(B) \leq 3$ . See Figure 6.1 for the OTD-code demonstrated with black vertices in the figure.

On the other hand, assume that  $C$  is an OTD-code of  $B$ . Since  $b_1$  and  $b_5$  are degree 1 vertices, their only neighbors, namely  $b_2$  and  $b_4$ , respectively, must be in  $C$  for the latter to be an open-dominating set of  $B$ . Moreover, at least one of  $b_1$  and  $b_3$  must be in  $C$  for  $b_2$  and  $b_5$  to be open-separated in  $B$ . Hence,  $|C| \geq 3$  and this establishes the result.  $\square$



Figure 6.2: Examples of  $G' \triangleright_q X$ , where  $G' \cong K_4$  and  $X \in \{P_4, Q_5\}$ . The vertex  $q$  (shaded) is obtained by identifying a vertex of  $G'$  and an articulation vertex of  $X$ .

This brings us to our result on the upper bound for OTD-numbers for block graphs.

**Theorem 6.1.** *Let  $G$  be a connected open-twin-free block graph with no isolated vertices. Moreover, let  $G$  neither be a copy of  $P_2$  nor of  $P_4$ . Moreover, let  $m_Q(G)$  be the number of non-leaf blocks of  $G$  with at least one non-articulation vertex. Then  $\gamma^{\text{OTD}}(G) \leq |V(G)| - m_Q(G) - 1$ .*

*Proof.* To start with, if  $G$  is a copy of the bull graph, then  $|V(G)| = 5$  and  $m_Q(G) = 1$ . Moreover, by Lemma 6.2,  $\gamma^{\text{OTD}}(G) = 3 = |V(G)| - m_Q(G) - 1$  and so, we are done. So, let us assume that  $G$  is not a copy of the bull graph.

We first choose a root block  $K_0 \in \mathcal{K}(G)$  according to the following two possibilities.

**Possibility 1:**  $G \cong G' \triangleright_q X$  for some block graph  $G'$  and some graph  $X$  that is a copy of either  $P_4$  or  $Q_5$ . In such a case, assume that  $x$  is the articulation vertex of  $X$  which is identified with the vertex  $q$  of  $G'$  to form  $G$ . Then, we choose  $K_0$  to be the block of  $G$  isomorphic to a  $P_2$  with vertices  $\{x, z\}$ , where  $z$  is the leaf vertex of  $X$  adjacent to  $x$ .

**Possibility 2:**  $G \not\cong G' \triangleright_q X$  for all block graphs  $G'$  and  $X$  that is a copy of either  $P_4$  or  $Q_5$ . In this case, choose  $K_0 \in \mathcal{K}(G)$  such that  $|V(K_0)| = \min\{|V(K)| : K \text{ is a leaf block of } G\}$ . Next, we construct a set  $C \subset V(G)$  by the following rules.

Rule 1: For every non-root leaf block  $K \in \mathcal{K}(G)$ , pick all vertices of  $K$  in  $C$ .

Rule 2: For every block  $K \in \mathcal{K}(G)$  that is either the root block  $K_0$  or is a non-leaf block in  $\mathcal{K}(G) \setminus \{K_0\}$ , (i) pick all articulation vertices of  $K$  in  $C$ ; and (ii) pick all but one non-articulation vertices of  $K$  in  $C$ .

To compute the cardinality of  $C$ , we note that, for the root block and every other non-leaf block  $K$  with at least one non-articulation vertex, exactly one vertex is left out from it in  $C$ . This gives  $|C| = |V(G)| - m_Q(G) - 1$ . Thus, the upper bound for  $\gamma^{\text{OTD}}(G)$  in the theorem is established on showing that  $C$ , indeed, is an OTD-code of  $G$ ; which is what we prove next.

To show that  $C$  is an OTD-code of  $G$ , we notice first of all that, if the root block  $K_0$  is isomorphic to  $P_2$ , then the (only) non-articulation vertex of  $K_0$  is open-dominated by the articulation vertex of  $K_0$  which belongs to  $C$ . In every other case when the root block is not isomorphic to  $P_2$ , all blocks  $K \in \mathcal{K}(G)$  have  $|V(K) \cap C| \geq 2$ . This makes  $C$  an open-dominating set of  $G$ . Now we show that  $C$  is also an open-separating set of  $V(G)$ . So, let us assume that  $u, v \in V(G)$  are distinct vertices of  $G$ . We now consider the following three cases.

► **Case 3:**  $u, v \in V(K)$  for some block  $K \in \mathcal{K}(G)$ .

We note here that, by the construction of  $C$ , for every block  $K \in \mathcal{K}(G)$ , at most one vertex of  $K$  is not in  $C$ . This implies that at least one of  $u$  and  $v$ , say  $u$  without loss of generality, must be in  $C$ . Then  $u$  open-separates  $u$  and  $v$  in  $G$ . ◀

► **Case 4:**  $u \in V(K)$ ,  $v \in V(K')$  for distinct  $K, K' \in \mathcal{K}(G)$  and both  $u, v \notin V(K) \cap V(K')$ .

In this case, let us first assume that  $V(K) \cap V(K') = \emptyset$ . Now, both  $K$  and  $K'$  cannot be root blocks of

$G$ . Without loss of generality, let us assume that  $K$  is not the root block of  $G$ . Since  $|V(K) \cap C| \geq 2$  by the construction of  $C$ , the block  $K$  has a vertex  $x$  in  $C$  other than  $u$ . Then,  $x$  open-separates  $u$  and  $v$  in  $G$ . Therefore, assume that  $V(K) \cap V(K') = \{w\}$  for some vertex  $w \in V(G)$ . Now, if either one of  $V(K)$  and  $V(K')$ , say  $V(K)$  without loss of generality, has cardinality at least 4, then  $|V(K) \cap C| \geq 3$  and so, at least one vertex in  $V(K) \setminus \{u, w\}$  belongs to  $C$  which open-separates  $u$  and  $v$  in  $G$ . So, assume that  $|V(K)| \leq 3$  and  $|V(K')| \leq 3$ . We next look at the following two subcases.

►► **Case 4.1:**  $|V(K)| = |V(K')| = 2$ .

Then, at least one of  $K$  and  $K'$  must be a non-leaf block, or else,  $G$  would have open-twins. So, without loss of generality, suppose that  $K$  is a non-leaf block. Then  $\{u\} = V(K) \cap V(K'')$  for some block  $K'' (\neq K, K') \in \mathcal{K}(G)$ . If however,  $K'$  is also a non-leaf-block, then  $\{v\} = V(K') \cap V(K''')$  for some block  $K''' (\neq K', K, K'') \in \mathcal{K}(G)$ . Now, at least one of  $K''$  and  $K'''$  is not the root block. Without loss of generality, therefore, assume that  $K''$  is not the root block. Then there is at least one vertex in  $V(K'') \setminus \{u\}$  which is in  $C$  and, hence, open-separates  $u$  and  $v$  in  $G$ . So, let us assume that  $K'$  is a leaf block, i.e.  $v$  is a non-articulation vertex of  $K'$ . Now again, if  $K''$  is not the root block, then there is at least one vertex in  $V(K'') \setminus \{u\}$  which is in  $C$  and, hence, open-separates  $u$  and  $v$  in  $G$ . So, now let us assume that  $K''$  is the root block. If  $G$  satisfies the conditions of Possibility 1, then  $|V(K'')| = 2$ . However, if  $G$  satisfies the conditions of Possibility 2, then again, since  $K'$  is a leaf block and  $|V(K')| = 2$ , by the minimality in order of the root block, we must have  $|V(K'')| = 2$ . So, assume  $z$  to be the non-articulation vertex of  $K_0$ . If  $P = G[z, u, w, v]$ , we have  $P \cong P_4$  with  $u$  and  $w$  being the articulation vertices and  $z$  and  $v$  being the leaf-vertices of  $P$ . Since  $G \not\cong P_4$ , we must have  $G \cong G' \triangleright_q P$  for some block graph  $G'$  and some vertex  $q \in \{u, w\}$  (note that both  $z$  and  $v$  are non-articulation vertices of  $G$ ). However, by the way we have chosen the root block  $K_0$ , we must have  $q = u$ . This implies that  $u$  is the negative articulation vertex of some  $K^* \in \mathcal{K}(G)$  such that  $K^* \notin \{K, K_0\}$ . This, in turn, implies that  $u$  and  $v$  are open-separated in  $G$  by some vertex in  $(V(K^*) \cap C) \setminus \{u\}$ . ◀◀

►► **Case 4.2:** At least one of  $V(K)$  and  $V(K')$  has cardinality 3.

Without loss of generality, let us assume that  $|V(K)| = 3$ . So, assume that  $K = G[w, u, y]$  for some vertex  $y \in V(G)$ . We must have  $y \notin C$  (otherwise  $y$  would open-separate  $u$  and  $v$ ). We first assume that  $K$  is the root block. If  $v$  is an articulation vertex of  $K'$ , then  $\{v\} = V(K') \cap V(K''')$  for some block  $K''' (\neq K') \in \mathcal{K}(G)$ . This implies that there exists a vertex in  $V(K''') \setminus \{v\}$  which open-separates  $u$  and  $v$  in  $G$ . Moreover, if  $v$  is a non-articulation vertex of  $K'$ , then we must have  $|V(K')| = 3$  (or else,  $K'$  is a leaf block of order smaller than the root block which is a contradiction). Assume that  $V(K') = \{w, v, a\}$  for some vertex  $a \in V(G)$ . If  $a$  is also a non-articulation vertex of  $K'$ , then  $K'$  is a leaf block that is not a root block and, hence,  $a \in C$ . If however,  $a$  is an articulation vertex of  $K'$ , then also,  $a \in C$ . Thus, either way,  $a$  open-separates  $u$  from  $v$  in  $G$ . Thus, we are done in the case that  $K$  is the root block of  $G$ .

So, let us now assume that  $K$  is *not* the root block of  $G$ . Now, if  $y \in C$ , then  $y$  open-separates  $u$  and  $v$  in  $G$ . So, let us assume that  $y \notin C$ , which implies that  $y$  is a non-articulation vertex of  $K$ . This, in turn, implies that  $u$  is an articulation vertex of  $K$ , or else,  $K$  would be a leaf block of  $G$  that is not the root block and so,  $y \in C$ , contrary to our assumption. So, let  $\{u\} = V(K) \cap V(K'')$  for some block  $K'' (\neq K) \in \mathcal{K}(G)$ . If however,  $v$  is also an articulation vertex of  $K'$ , then  $\{v\} = V(K') \cap V(K''')$  for some block  $K''' (\neq K') \in \mathcal{K}(G)$ . Now, at least one of  $K''$  and  $K'''$  is not the root block. So, without loss of generality, assume that  $K''$  is not the root block. Then, there is at least one vertex in  $V(K'') \setminus \{u\}$  which is in  $C$  and, hence, open-separates  $u$  and  $v$  in  $G$ . So, let us assume that  $v$  is a non-articulation vertex of  $K'$ . Again, if  $K''$  is not the root block, then there exists at least one vertex in  $V(K'') \setminus \{u\}$  which is in  $C$  and, hence, open-separates  $u$  and  $v$  in  $G$ . So, now assume  $K''$  to be the root block. If  $|V(K'')| \geq 3$ , then there exists a vertex of  $V(K'') \setminus \{u\}$  in  $C$  and, hence, open-separates  $u$  and  $v$  in  $G$ . So, let us assume that  $|V(K'')| = 2$  and that  $z$  is the non-articulation of  $K''$ . If  $|V(K')| = 3$ , then suppose that  $V(K') = \{w, v, a\}$  for some vertex  $a \in V(G)$ . If  $a$  is also a non-articulation vertex of  $K'$ , then  $K'$  is a leaf block that is not a root block and hence,  $a \in C$ . If however,  $a$  is an articulation vertex of  $K'$ , then also,  $a \in C$ . Thus, either way,  $a$  open-separates  $u$

from  $v$  in  $G$  and we are done in the case that  $|V(K')| = 3$ . So, let us finally assume that  $|V(K')| = 2$  and that  $v \in V(K')$  is a non-articulation vertex of  $K'$ .

If  $B = G[z, u, y, w, v]$ , we have  $B \cong Q_5$  with  $u$  and  $w$  being the articulation vertices and  $z$  and  $v$  being the leaf-vertices of  $B$ . Since by our assumption,  $G \not\cong Q_5$ , we have  $G \cong G' \triangleright_q B$  for some block graph  $G'$  and some vertex  $q \in \{u, w\}$  (note that  $z, y$  and  $v$  are non-articulation vertices of  $G$ ). Now, by the way we have chosen the root block, this implies that we must have  $q = u$ . This further implies that  $u$  is the negative articulation vertex of  $K^*$  for some  $K^* \in \mathcal{K}(G)$  such that  $K^* \notin \{K, K_0\}$ . Hence,  $u$  and  $v$  are open-separated in  $G$  by some vertex in  $V(K^*) \cap C \setminus \{u\}$ . ◀◀

Thus, the vertices  $u$  and  $v$  are open-separated in this case as well. ◀

This, therefore, proves that  $C$  is an OTD-code of  $G$  and with that, we prove the theorem. ◻

Applying Theorem 6.1 to each component of a block graph, one has the following general result.

**Corollary 6.1.** *Let  $G$  be an open-twin-free block graph with  $k$  components and no isolated vertices. Moreover, let no component of  $G$  be either a copy of  $P_2$  or of  $P_4$ . Also, let  $m_Q(G)$  be the number of non-leaf blocks of  $G$  with at least one non-articulation vertex. Then  $\gamma^{\text{OTD}}(G) \leq |V(G)| - m_Q(G) - k$ .*

Foucaud et al. [89] have shown that, for any open-twin-free graph  $G$ ,  $\gamma^{\text{OTD}}(G) \leq |V(G)| - 1$  unless  $G$  is a half-graph. Noting that  $P_2$  and  $P_4$  are the only block graphs that are half-graphs, Theorem 6.1 can be seen as a refinement of this result for block graphs.

We now show that the upper bound given in Theorem 6.1 is tight and is attained by arbitrarily large connected block graphs. To prove so, for two non-negative integers  $k$  and  $l$  such that  $k + l \geq 2$ , let us define a subfamily  $G_{k,l}$  of block graphs by the following rule: Let  $T_1, T_2, \dots, T_k$  be  $k$  copies of  $K_3$  with  $V(T_i) = \{u_i, v_i, w_i\}$  for each  $1 \leq i \leq k$ . Further, let  $A_1, A_2, \dots, A_k$  be  $k$  copies of  $P_2$  with  $V(A_i) = \{a_i, b_i\}$  for each  $1 \leq i \leq k$ , and  $L_1, L_2, \dots, L_l$  be  $l$  copies of  $P_3$  with  $V(L_j) = \{x_j, y_j, z_j\}$  for each  $1 \leq j \leq l$ . Let  $G_{k,l}$  be the graph obtained by identifying the vertices  $v_i$  with  $b_i$  for each  $1 \leq i \leq k$  and identifying the vertices  $u_i$  and  $z_j$  for all  $1 \leq i \leq k$  and  $1 \leq j \leq l$  into a single vertex  $u$ , say. See Figure 6.3 for an example of a construction of  $G_{k,l}$  with  $k = 2$  and  $l = 3$ . As a matter of reference, we continue to call the vertices of  $G_{k,l}$  obtained by identifying  $v_i$  with  $b_i$ , for all  $1 \leq i \leq k$ , as  $v_i$  itself.

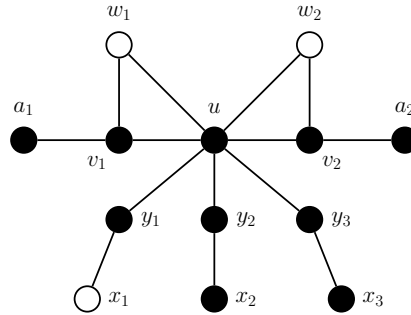


Figure 6.3: Graph  $G_{2,3}$  whose OTD-number attains the upper bound in Theorem 6.1. The black vertices represent those included in the OTD-code  $C$  of  $G$  as described in the proof of Theorem 6.1.

**Proposition 6.1.** *For all positive integers  $k$  and  $l$  with  $l + k \geq 2$ ,  $\gamma^{\text{OTD}}(G_{k,l}) = |V(G_{k,l})| - m_Q(G_{k,l}) - 1$ .*

*Proof.* First of all, we have  $|V(G_{k,l})| = 3k + 2l + 1$ , and  $m_Q(G_{k,l}) = k$ . Therefore, by Theorem 6.1, we have  $\gamma^{\text{OTD}}(G_{k,l}) \leq 2(k + l)$ .

Now, let  $C$  be an OTD-code of  $G_{k,l}$ . For  $C$  to be an open-dominating set of  $G_{k,l}$ , the only neighbors  $v_i$  and  $y_j$  of the degree 1 vertices  $a_i$  and  $x_j$ , respectively, of  $G_{k,l}$  must be in  $C$ . Moreover,  $u \in C$

for each pair of  $w_i$  and  $a_i$  to be open-separated in  $G_{k,l}$  by  $C$ . Similarly, at least all but one of the  $x_i$ 's must belong to  $C$  for each pair of vertices  $y_i$  and  $y_j$ , for  $1 \leq i < j \leq l$ , to be open-separated in  $G_{k,l}$  by  $C$ . Let us first assume that  $x_2, x_3, \dots, x_l$  belong to  $C$ , without loss of generality. Then, for all  $1 \leq i \leq k$ , at least one of  $w_i$  and  $a_i$  must be in  $C$  for the pair  $y_1$  and  $v_i$ , for  $1 \leq i < j \leq k$ , to be open-separated in  $G_{k,l}$  by  $C$ . Adding up, therefore, we have  $|C| \geq 2(k+l)$ . On the other hand, if  $x_1, x_2, \dots, x_l \in C$ , then, to open-separate the vertices  $v_i$  and  $v_j$  (for  $1 \leq i < j \leq k$ ), we need at least one of  $w_i$  and  $a_i$  (for  $k-1$  such  $i$ ) to belong to  $C$ . Then again, the count adds up to  $|C| \geq 2(k+l)$ . This proves the proposition.  $\square$

### 6.1.1.2 Lower bounds on OTD-numbers of block graphs

We now turn to studying lower bounds on the OTD-numbers of block graphs. We find the lower bounds of OTD-numbers both in terms of the number of vertices and in terms of the number of blocks of a block graph. For the rest of this section, given a block graph  $G$ , by  $\mathcal{K}_{leaf}(G)$  we denote the set of all leaf blocks of  $G$  with at least one edge in the block. Moreover, by the symbol  $n_i(G)$ , we denote the number of vertices of degree  $i$  in the graph  $G$ .

#### Lower bound in terms of the order of the graph

In terms of the order of the graph, we prove the following lower bound on the OTD-number of an OTD-admissible block graph.

**Theorem 6.2.** *Let  $G$  be a connected open-twin-free block graph. Then we have*

$$\gamma^{OTD}(G) \geq \frac{|V(G)| + 2}{3}.$$

To prove the theorem, we first go through some definitions and establish some lemmas. To that end, we also recall some definitions and the statements of some of the lemmas and corollaries that we had already encountered in Section 4.1.2 and which will again be used in this section.

**Lemma 4.4.** *Let  $G$  be a connected block graph with at least one edge. Then we have*

$$|\mathcal{K}(G)| \leq |V(G)| - 1 - |\mathcal{K}_{leaf}(G)| + n_1(G).$$

**Corollary 4.3.** *Let  $G$  be a block graph with  $k$  components. Then we have*

$$|\mathcal{K}(G)| - n_0(G) \leq |V(G)| - k.$$

**Definition 4.1.** For a given code  $C$  (not necessarily an OTD-code) of a connected block graph  $G$ , let us assume that the sets  $C_1, C_2, \dots, C_k$  partition the code  $C$  such that the induced subgraphs  $G[C_1], G[C_2], \dots, G[C_k]$  of  $G$  are the  $k$  components of the subgraph  $G[C]$  of  $G$  induced by  $C$ . Note that each  $C_i$  is a block graph (since every induced subgraph of a block graph is also a block graph). Then, the vertex set  $V(G)$  is partitioned into the following four parts.

- (1)  $V_1 = C$ ,
- (2)  $V_2 = \{v \in V(G) \setminus V_1 : |N_G(v) \cap C| = 1\}$ ,
- (3)  $V_3 = \{v \in V(G) \setminus V_1 : \text{there exist distinct } i, j \leq k \text{ such that } N_G(v) \cap C_i \neq \emptyset \text{ and } N_G(v) \cap C_j \neq \emptyset\}$ ,
- (4)  $V_4 = V(G) \setminus (V_1 \cup V_2 \cup V_3)$ . Note that, for all  $v \in V_4$ , we have  $N_G(v) \cap C \subset C_i$  for some  $i$  and that  $|N_G(v) \cap C_i| \geq 2$ .

**Lemma 4.7.** *Let  $G$  be a connected block graph,  $C$  be a code (not necessarily an OTD-code) of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Then, we have  $|V_3| \leq k - 1$ .*

We now prove a series of lemmas establishing upper bounds on the orders of the vertex subsets  $V_2$  and  $V_4$  of a connected block graph  $G$ .

**Lemma 6.3.** *Let  $G$  be a connected open-twin-free block graph and  $C$  be an OTD-code of  $G$ . Then, we have  $|V_2| \leq |C| - n_1(G[C])$ .*

*Proof.* By definition of  $V_2$ , each vertex  $v \in V_2$  has a unique neighbor  $u$  in  $C$ , that is,  $N_G(v) \cap C = \{u\}$ . Hence, there can be at most  $|C|$  vertices in  $V_2$ . Moreover, since  $C$  is an OTD-code, if  $u$  has a neighbor  $w \in N_G(u) \cap C$  such that  $\deg_{G[C]}(w) = 1$ , then  $v$  and  $w$  are not open-separated. Thus, there are at most  $|C| - n_1(G[C])$  vertices in  $V_2$ .  $\square$

**Lemma 6.4.** *Let  $G$  be a connected open-twin-free block graph,  $C$  be an OTD-code of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Then, we have  $|V_4| \leq |\mathcal{K}(G[C])| \leq |C| - 2k + n_1(G[C])$ .*

*Proof.* Let  $v$  be any vertex in  $V_4$  and let  $G[C_i]$  be the component of  $G[C]$  such that  $N_G(v) \cap C \subseteq C_i$ . Moreover,  $|N_G(v) \cap C_i| \geq 2$ . Then, notice that  $N_G(v) \cap C$  must be a subset of exactly one block of  $G[C_i]$ , or else,  $G[C_i]$  would be disconnected, as  $v \notin C$ . This implies that  $|V_4| \leq |\mathcal{K}(G[C])| - n_0(G[C]) \leq |C| - k$ , by Corollary 4.3.

Since  $C$  is an OTD-code of  $G$ , we have  $n_0(G[C]) = 0$ . So, assume that  $G[C_i]$  is a component of  $G[C]$  with at least one edge. Then, by Lemma 4.4, we have  $|\mathcal{K}(G[C_i])| \leq |C_i| - 1 - |\mathcal{K}_{leaf}(G[C_i])| + n_1(G[C_i]) \leq |C_i| - 2 + n_1(G[C_i])$ . This implies that

$$|V_4| \leq |\mathcal{K}(G[C])| = \sum_{1 \leq i \leq k} |\mathcal{K}(G[C_i])| \leq \sum_{1 \leq i \leq k} (|C_i| - 2 + n_1(G[C_i])) = |C| - 2k + n_1(G[C]).$$

$\square$

This brings us to the proof of Theorem 6.2.

**Theorem 6.2.** *Let  $G$  be a connected open-twin-free block graph. Then we have*

$$\gamma^{\text{OTD}}(G) \geq \frac{|V(G)| + 2}{3}.$$

*Proof.* Let  $|V(G)| = n$ . Assume  $C$  to be an OTD-code of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  to be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . Recalling from Definition 4.1 the sets  $V_1, V_2, V_3, V_4$  that partition  $V(G)$ , we prove the theorem using the relation  $|V(G)| = |C| + |V_2| + |V_3| + |V_4|$  and the upper bounds for  $|V_2|$ ,  $|V_3|$  and  $|V_4|$  in Lemmas 6.3, 4.7 and 6.4, respectively. Therefore, we have

$$\begin{aligned} n &= |C| + |V_2| + |V_3| + |V_4| \\ &\leq |C| + |C| - n_1(G[C]) + k - 1 + |C| - 2k + n_1(G[C]) \\ &= 3|C| - k - 1 \leq 3|C| - 2, \quad \text{using } k \geq 1. \end{aligned}$$

Hence, the result holds.  $\square$

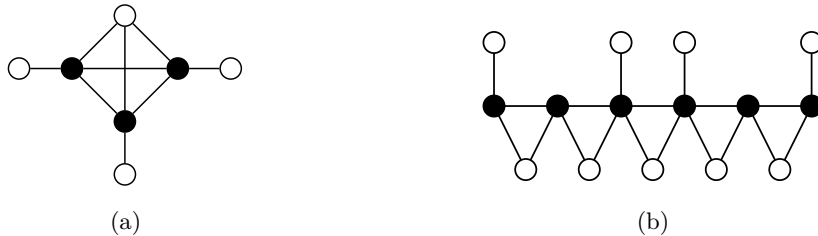


Figure 6.4: Extremal cases where the lower bounds in Theorem 6.2 are attained. The black vertices form a minimum OTD-code. Figure (a) represents the graph  $Z$  described in Proposition 6.2.

We now look at examples of connected block graphs that are extremal with respect to Theorem 6.2. To that end, we prove the following proposition.



**Proposition 6.2.** (1) Consider the block graph  $Z$  consisting of a clique on four vertices, of which three are support vertices to one leaf each (see Figure 6.4a). Then  $Z$  is the only block graph whose OTD-number attains the bound in Theorem 6.2.

(2) For any  $G \not\cong Z$ , the bound for the OTD-number in Theorem 6.2 becomes

$$\gamma^{\text{OTD}}(G) \geq \frac{|V(G)|}{3} + 1.$$

In this case, there are arbitrarily large connected block graphs whose OTD-number attains this lower bound.

*Proof.* For any block graph  $G$  whose OTD-number attains the bound in Theorem 6.2, we must have equalities in the equations in the proof of Theorem 6.2 when  $C$  is an OTD-code. This implies that we must have  $k = 1$ . Moreover, we must have  $|V_2| = |C| - n_1(G[C])$ ,  $|V_3| = 0$  and

$$|V_4| = |C| - 2 + n_1(G[C]) = |\mathcal{K}(G[C])| \quad (\text{using Lemma 6.4 and } k = 1)$$

(1) The last equation stems from an equality in the statement of Lemma 4.4, which implies that  $|\mathcal{K}_{\text{leaf}}(G[C])| = 1$ . This implies that  $C$  is a clique. Since  $C$  is an OTD-code, we must have  $|C| \geq 2$ . If  $|C| = 2$ , then we have  $|V_4| = n_1(G[C]) = 1$ . This is a contradiction, as  $|C| = 2$  implies that  $G[C] \cong P_2$  and hence,  $n_1(G[C]) = 2$ . Thus, we have  $|C| \geq 3$ . This implies that  $n_1(G[C]) = 0$  and so, we get  $|V_4| = |C| - 2 = 1 \implies |C| = 3$ . Moreover,  $|V_2| = 3$  determines the graph to be  $Z$ .

(2) We now assume that  $G \not\cong Z$ . If  $k \geq 2$ , then putting so in the proof of Theorem 6.2, we get  $|V(G)| \leq 3|C| - 3$  and hence, the result follows. Therefore, for the rest of this proof, let us assume that  $k = 1$ . In other words,  $G[C]$  is a connected subgraph of  $G$ . Now, we must have  $|\mathcal{K}_{\text{leaf}}(G[C])| \geq 2$ , or else, if  $|\mathcal{K}_{\text{leaf}}(G[C])| = 1$ , the same analysis as in (1) would follow and we would have  $G \cong Z$ , a contradiction. Thus, putting  $|\mathcal{K}_{\text{leaf}}(G[C])| \geq 2$  in the proof of Lemma 6.4, we get  $|V_4| \leq |\mathcal{K}(G[C])| \leq |C| - 3k + n_1(G[C])$ . Feeding this further in the proof of Theorem 6.2 when  $C$  is an OTD-code, we get  $|V(G)| \leq 3|C| - 3$  and hence, the lower bound holds.

To show that there exist arbitrarily large and infinitely many block graphs whose OTD-numbers attain this bound, for  $\ell \geq 6$ , we consider a path on vertices  $u_1, u_2, \dots, u_\ell$ . For  $i = 1, \ell$  and  $3 \leq i \leq \ell - 2$ , attach a vertex  $v_i$  by the edge  $v_i u_i$ . Moreover, for each pair  $u_i, u_{i+1}$  for  $1 \leq i \leq \ell - 1$ , attach a vertex  $w_i$  by edges  $w_i u_i$  and  $w_i u_{i+1}$ . We call the graph  $G_\ell$ . See Figure 6.4b for an example with  $\ell = 6$ . Then it can be verified that  $C = \{u_1, u_2, \dots, u_\ell\}$  is a minimum OTD-code of  $G_\ell$ . Since  $|V(G_\ell)| = 3\ell - 3$ , the OTD-number of  $G_\ell$  attains the lower bound.  $\square$

## Lower bound in terms of the number of blocks

With respect to the number of block, we prove the following lower bound on the OTD-number of an OTD-admissible block graph.

**Theorem 6.3.** Let  $G$  be a connected open-twin-free block graph and let  $\mathcal{K}(G)$  be the set of all blocks of  $G$ . Then we have

$$\gamma^{\text{OTD}}(G) \geq \frac{|\mathcal{K}(G)|}{2} + 1.$$

To prove the theorem, we again recall some definitions and lemmas (without proof) established in Section 4.1.2.

**Definition 4.3.** Let  $C$  be a code (not necessarily an OTD-code) of  $G$  and  $G[C_1], G[C_2], \dots, G[C_k]$  be all the components of  $G[C]$ , where  $C_1, C_2, \dots, C_k$  are subsets of  $C$ . First, we define  $\mathcal{I}_G(C) = \{K \in \mathcal{K}(G) : V(K) \subset V(C) \text{ for some } L \in \mathcal{K}(G[C])\}$ . Moreover, for each  $1 \leq i \leq k$ , let  $\mathcal{I}_G(C_i) = \{K \in \mathcal{K}(G) : V(K) \subset V(C) \text{ for some } L \in \mathcal{K}(G[C_i])\}$ . Next, we define the following types of blocks of  $G$ .

(1) Let  $\mathcal{K}_C(G) = \{K \in \mathcal{I}_G(C) : V(K) \subset C\}$ , i.e. all blocks of  $G$  which are also blocks of the subgraph  $G[C]$  (also a block graph) of  $G$ .

- (2) Let  $\mathcal{K}_{\overline{C}}(G) = \mathcal{K}(G) \setminus \mathcal{I}_G(C)$ . In other words, the set  $\mathcal{K}_{\overline{C}}(G)$  includes all blocks of  $G$  which do not contain any vertices of the code  $C$ .
- (3) For  $i = 2, 3, 4$ , let  $\mathcal{K}_i(G) = \{K \in \mathcal{I}_G(C) : V(K) \cap V_i \neq \emptyset\}$  (recall the sets  $V_1, V_2, V_3, V_4$  from Definition 4.1).

**Lemma 4.9.**  $|\mathcal{K}_2(G)| \leq |V_2|$ .

**Lemma 4.10.**  $|\mathcal{K}_3(G)| \leq 2(k - l)$ .

**Lemma 4.11.**  $|\mathcal{K}_{\overline{C}}(G)| \leq l - 1$ .

**Lemma 4.12.**  $|\mathcal{K}_C(G) \cup \mathcal{K}_4(G)| \leq |\mathcal{K}(G[C])| - n_0(G[C])$ .

We now prove Theorem 6.3.

**Theorem 6.3.** *Let  $G$  be a connected open-twin-free block graph and  $\mathcal{K}(G)$  be the set of all blocks of  $G$ . Then we have*

$$\gamma^{\text{OTD}}(G) \geq \frac{|\mathcal{K}(G)|}{2} + 1.$$

*Proof.* Let  $C$  be an OTD-code of the block graph  $G$ . Then, we have

$$\mathcal{K}(G) = (\mathcal{K}_C(G) \cup \mathcal{K}_4(G)) \cup \mathcal{K}_{\overline{C}}(G) \cup \mathcal{K}_2(G) \cup \mathcal{K}_3(G).$$

Therefore, using Lemma 4.9, 4.10, 4.11 and 4.12, we have

$$\begin{aligned} |\mathcal{K}(G)| &\leq |\mathcal{K}_C(G) \cup \mathcal{K}_4(G)| + |\mathcal{K}_{\overline{C}}(G)| + |\mathcal{K}_2(G)| + |\mathcal{K}_3(G)| \\ &\leq |\mathcal{K}(G[C])| - n_0(G[C]) + l - 1 + |V_2| + 2(k - l) \\ &\leq |C| - 2k + l - 1 + |C| + 2(k - l) \quad [\text{using Lemmas 6.3 and 6.4 and } n_0(G[C]) = 0] \\ &= 2|C| - l - 1 \\ &\leq 2|C| - 2, \quad \text{using } l \geq 1. \end{aligned}$$

This proves the theorem.  $\square$

Note that, for any tree  $G$  (which are particular block graphs with each block being of order 2), we have  $|\mathcal{K}(G)| = |E(G)| = |V(G)| - 1$ . The lower bound for OTD-numbers of trees given by Theorem 6.3 is the same when  $|V(G)|$  is even and one short when  $|V(G)|$  is odd as the lower bound given in [190] (which is  $\lceil \frac{|V(G)|}{2} \rceil + 1$ ). However, for  $|V(G)|$  odd, this lower bound for OTD-numbers given by Theorem 6.3 is tight with the graph  $Z$  in Figure 6.4a attaining this bound. Apart from this example, there are infinite families of trees reaching the bound in Theorem 6.3 (see [190]).

## 6.1.2 $n$ -cycles

In this section, we establish the exact values of the OTD-numbers of  $n$ -cycles in terms of their order, that is,  $n$ . Exact OTD-numbers of cycles have been studied in [9]. However, the result for the even order of cycles in the said work needed to be corrected and as such, we state and prove the result in its entirety as follows.

The results of this section have appeared in [26].

**Theorem 6.4.** *For any cycle  $C_n$  on  $n$  vertices such that  $n \geq 3$  and  $n \neq 4$ , we have*

$$\gamma^{\text{OTD}}(C_n) = \begin{cases} \lceil \frac{2n}{3} \rceil, & \text{for odd } n; \\ 2 \lceil \frac{n}{3} \rceil, & \text{for even } n. \end{cases}$$

*Proof.* We prove the theorem by first showing that  $\lceil \frac{2n}{3} \rceil$  for odd  $n$  and  $2\lceil \frac{n}{3} \rceil$  for even  $n$  is a lower bound on  $\gamma^{\text{OTD}}(C_n)$  and then providing an OTD-code of  $C_n$  of exactly the same cardinality as the lower bound. We start with establishing the lower bound first.

Seo and Slater showed in [190] that if  $G$  is a regular graph on  $n$  vertices, of regular-degree  $r$  and with no open twins, then we have  $\gamma^{\text{OTD}}(G) \geq \frac{2}{1+r}n$ . Using this result in [190] for the cycle  $C_n$ , therefore, we have  $\gamma^{\text{OTD}}(C_n) \geq \frac{2}{3}n$ , that is,  $\gamma^{\text{OTD}}(C_n) \geq \lceil \frac{2n}{3} \rceil$ . Now, for  $n \neq 6k + 4$  for any non-negative integer  $k$ , we have

$$\left\lceil \frac{2n}{3} \right\rceil = \begin{cases} \left\lceil \frac{2n}{3} \right\rceil, & \text{for odd } n; \\ 2\left\lceil \frac{n}{3} \right\rceil, & \text{for even } n. \end{cases}$$

Thus, the only case left to prove is the following claim.

■ **Claim 1.** For  $n = 6k + 4$  with  $k \geq 1$ , we have  $\gamma^{\text{OTD}}(C_n) \geq 2\lceil \frac{n}{3} \rceil = 4k + 4$ .

*Proof of claim.* The proof of the last claim is by induction on  $k$  with the base case being for  $k = 1$ , that is, when  $C_n$  is a cycle on  $n = 10$  vertices. We first show the result for  $n = 10$ .

■■ **Claim 1.1.**  $\gamma^{\text{OTD}}(C_{10}) \geq 8$ .

*Proof of claim.* Let  $V(C_{10}) = \{v_1, v_2, \dots, v_{10}\}$  and  $S$  be a minimum OTD-code of  $C_{10}$ . Then,  $|S| < 10$  by a characterization result by Foucaud et al. [89] on the extremal graphs  $G$  for which  $\gamma^{\text{OTD}}(G) = |V(G)|$ . Hence, there exists a vertex  $v_1$  (without loss of generality) such that  $v_1 \notin S$ . We consider the induced 5-paths  $P_1 : v_1v_2v_3v_4v_5$  and  $P_2 : v_1v_{10}v_9v_8v_7$ . Then, from the path  $P_1$ , the vertex  $v_3 \in S$  for the latter to total-dominate  $v_2$  and the vertex  $v_5 \in S$  for the latter to open-separate the pair  $v_2, v_4$ . By the same argument, from path  $P_2$ , the vertices  $v_9, v_7 \in S$ . Moreover, at least one vertex from each of the pairs  $(v_2, v_4), (v_4, v_6), (v_6, v_8), (v_8, v_{10}), (v_{10}, v_2)$  must belong to  $S$  for the latter to total-dominate  $v_3, v_5, v_7, v_9, v_1$ , respectively. Hence, the result follows from counting. ■■

Thus, the result holds for the base case of the induction hypothesis. We, therefore, assume  $k \geq 2$  and that  $\gamma^{\text{OTD}}(C_m) \geq 4q + 4$  for all cycles  $C_m$  with  $|V(C_m)| = 6q + 4$  and  $q \in \{1, 2, \dots, k-1\}$ . Toward contradiction, let us assume that  $\gamma^{\text{OTD}}(C_n) < 4k + 4$ . Moreover, let  $V(C_n) = \{v_1, v_2, \dots, v_n\}$ . Then again, by the characterization result in [89], we have  $\gamma^{\text{OTD}}(C_n) < n$ . This implies that, for any minimum OTD-code  $S$  of  $C_n$ , there exists a pair  $(v_{n-6}, v_{n-5})$  (by a possible renaming of vertices) such that  $v_{n-6} \in S$  and  $v_{n-5} \notin S$ . Let  $C'_{n-6}$  be the cycle on  $n-6$  vertices formed by adding the edge  $v_1v_{n-6}$  in the graph  $C_n - \{v_{n-5}, v_{n-4}, \dots, v_n\}$ . Note that  $|V(C'_{n-6})| = 6(k-1) + 4$  and hence, the induction hypothesis applies to it to give

$$\gamma^{\text{OTD}}(C'_{n-6}) \geq 4(k-1) + 4 = 4k. \quad (6.1)$$

Now, let  $S' = S - \{v_{n-5}, v_{n-4}, \dots, v_n\}$ .

■■ **Claim 1.2.**  $S'$  is an OTD-code of  $C'_{n-6}$ .

*Proof of claim.* To show that  $S'$ , first of all, is a total-dominating set of  $C'_{n-6}$ , we notice that the vertices  $v_1, v_5 \notin S$ . Therefore, all vertices in the set  $\{v_2, v_3, \dots, v_{n-6}\}$  remain total-dominated by  $S'$ . Moreover,  $S'$  total-dominates  $v_1$  by virtue of  $v_{n-6} \in S'$ . This proves that  $S'$  is a total-dominating set of  $C'_{n-6}$ .

We now show that  $S'$  is also a total-separating set of  $C'_{n-6}$ . To that end, since  $v_{n-5} \notin S$ , if now the vertex  $v_n \notin S$  as well, then  $S'$  clearly open-separates every pair of vertices in  $C'_{n-6}$  and hence, is an OTD-code. If however,  $v_n \in S$  and open-separates a pair of vertices in  $C'_{n-6}$ , the pair can either be  $(v_1, v_2)$  or  $(v_1, v_3)$ . Since  $n \geq 16$ , we have  $3 < n-6$  and hence,  $v_{n-6} \in S'$  open-separates the pairs  $(v_1, v_2)$  and  $(v_1, v_3)$  in  $C'_{n-6}$ . Therefore,  $S'$  is an OTD-code of  $C'_{n-6}$ . ■■

■ ■ **Claim 1.3.**  $|S \cap \{v_{n-5}, v_{n-4}, \dots, v_n\}| \geq 4$ .

*Proof of claim.* Since  $v_{n-6} \notin S$ , it implies that  $v_{n-3} \in S$  in order for the latter to total-dominate the vertex  $v_{n-4}$ . Moreover, we also have  $v_{n-1} \in S$  in order for  $S$  to open-separate the pair  $(v_{n-2}, v_{n-4})$ . Furthermore, we must have at least one vertex each from the pairs  $(v_{n-4}, v_{n-2})$  and  $(v_{n-2}, v_n)$  belonging to  $S$  in order for the latter to total-dominate the vertices  $v_{n-3}$  and  $v_{n-1}$ , respectively. Finally, at least one vertex from the pair  $(v_{n-4}, v_n)$  must also belong to  $S$  in order for  $S$  to open-separate the pair  $(v_{n-3}, v_{n-1})$ . This proves that the result holds. ■ ■

Recall that  $|S| = \gamma^{\text{OTD}}(C_n) < 4k + 4$ , by assumption. Thus, we have

$$\gamma^{\text{OTD}}(C'_{n-6}) \leq |S'| = |S| - |S \cap \{v_{n-5}, v_{n-4}, \dots, v_n\}| < 4k + 4 - 4 = 4k,$$

a contradiction to the Inequality (7.1). This proves the claim and establishes the lower bound on  $\gamma^{\text{OTD}}(C_n)$ . ■

The theorem is, therefore, proved by providing an OTD-code  $S$  of  $C_n$  of the exact same cardinality as the lower bound, that is,

$$|S| = \begin{cases} \lceil \frac{2n}{3} \rceil, & \text{for odd } n, \\ 2 \lfloor \frac{n}{3} \rfloor, & \text{for even } n. \end{cases} \quad (6.2)$$

Let  $V(C_n) = \{v_1, v_2, \dots, v_n\}$  and that  $n = 6k + r$ , where  $r \in \{0, 1, 2, 3, 4, 5\}$ . For  $k = 0$ , that is,  $C_n$  being either a 3-cycle or a 5-cycle, it can be checked that the sets  $\{v_1, v_2\}$  and  $\{v_1, v_2, v_3, v_4\}$  are the respective OTD-codes. Thus, the result holds in this case. For the rest of this proof, therefore, we assume that  $n \geq 6$ , that is,  $k \geq 1$ . We now construct a vertex subset  $S$  of  $C_n$  by including the following vertices in it (see Figure 6.5).

- (1)  $v_{6i-4}, v_{6i-3}, v_{6i-1}, v_{6i}$  for all  $i \in \{1, 2, \dots, k\}$  if  $r = 0, 3$ . In this case, we have  $|S| = 4k$  for  $r = 0$  and  $|S| = 4k + 2$  for  $r = 3$ .
- (2)  $v_{6i-4}, v_{6i-3}, v_{6i-2}, v_{6i-1}$  for all  $i \in \{1, 2, \dots, k\}$  if  $r \neq 0$ ; with
  - (a) the vertices  $v_{6k}, v_{6k+1}, \dots, v_{6k+r-1}$  if  $r = 1, 2, 4$ . In this case, we have  $|S| = 4k + r$ ; and
  - (b) the vertices  $v_{6k+1}, v_{6k+2}, v_{6k+3}, v_{6k+4}$  if  $r = 5$ . In this case, we have  $|S| = 4k + 4$ .

It can be checked that the constructed set  $S$  is, indeed, an OTD-code of  $C_n$  and of the cardinality as in Equation (6.2). This proves the result. □

### 6.1.3 $C_4$ -free graphs with bounded maximum degree

In this section, we study a tight upper bound of the OTD-numbers of  $C_4$ -free graphs in terms of the order and the maximum degree of the graphs. Upper bounds on the OTD-number of graphs  $G$  of order  $n$  with  $\gamma^{\text{OTD}}(G) = n$  have been characterized by Foucaud et al. in [89] as the family of half-graphs. In [191], Seo and Slater characterized the trees  $T$  of order  $n$  with  $\gamma^{\text{OTD}}(T) = n - 1$ . Such upper bounds have also been studied for classic identifying codes, where the open neighborhood is replaced by the closed neighborhood in their definition. In particular, some works have explored the best possible upper bound for the optimal order of an identifying code, depending on the maximum degree and the order of the graph [43, 41, 42, 99, 104]. Those works have inspired our current research.

Further motivation comes from the work by Henning and Yeo [130], who initiated the study of upper bounds on the OTD-number of graphs of given maximum degree, focusing on regular graphs.

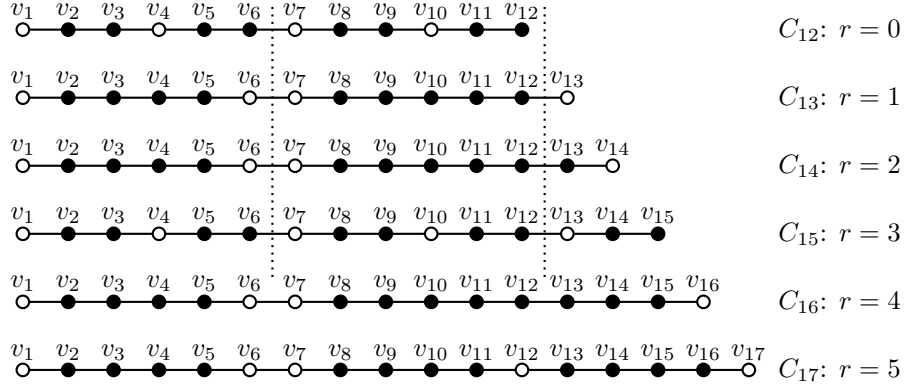


Figure 6.5:  $C_n$  (by joining the vertices  $v_1$  and  $v_n$  by an edge in each case). The set of black vertices represent an OTD-code.

They proved, using an interplay with transversal of hypergraphs, that  $\gamma^{\text{OTD}}(G) \leq \frac{3}{4}n$  for every open-twin-free connected cubic graph  $G$  and this bound is tight for the complete graph  $K_4$  and the hypercube of dimension 3. They also suggested that perhaps, more generally,  $\gamma^{\text{OTD}}(G) \leq \left(\frac{\Delta}{\Delta+1}\right)n$  holds for every connected open-twin-free  $\Delta$ -regular graph  $G$  of order  $n$ . Our goal is to investigate what happens when the graph is non-regular. We will see that, interestingly, the above conjectured bound for the OTD-number of regular graphs must be modified for non-regular graphs.

All results of this section have appeared in [40].

### 6.1.3.1 Notations, terminologies and the main result

Let  $G$  be a connected graph of order  $n \geq 2$  that is open-twin-free. If  $n = 2$ , then  $G = P_2$  and  $\gamma^{\text{OTD}}(G) = 2 = n$ . If  $n = 3$ , then  $G = K_3$  and  $\gamma^{\text{OTD}}(G) = 2 = \frac{2}{3}n$ . If  $n = 4$ , then  $G = P_4$  and  $\gamma^{\text{OTD}}(G) = 4 = n$ . Our aim is to obtain a best possible upper bound on OTD-numbers of open-twin-free connected graphs  $G$  of order  $n \geq 5$  that contains no 4-cycles and have bounded maximum degree.

For  $\Delta \geq 3$ , a *subdivided star*  $T_\Delta = S(K_{1,\Delta})$  is the tree obtained from a star  $K_{1,\Delta}$  with universal vertex  $v$ , say, by subdividing every edge exactly once. To emphasize the fact that a subdivided star has been obtained from a  $\Delta$ -star, we may sometimes call it a  $\Delta$ -*subdivided star*. The resulting tree  $T_\Delta$  has order  $n = 2\Delta + 1$  and maximum degree  $\Delta(T_\Delta) = \Delta$ . For example, when  $\Delta = 4$  the subdivided star  $T_4 = S(K_{1,4})$  is illustrated in Figure 6.6(a). For  $\Delta \geq 3$ , a *reduced subdivided star*  $T_\Delta^*$  is the tree obtained from a subdivided star  $K_{1,\Delta}$  by removing exactly one leaf. The resulting tree  $T_\Delta^*$  has order  $n = 2\Delta$  and maximum degree  $\Delta(T_\Delta^*) = \Delta$ . For example, when  $\Delta = 4$  the reduced subdivided star  $T_4^*$  is illustrated in Figure 6.6(b). In both the subdivided and the reduced subdivided tree, the vertex  $v$  is called the *central vertex* of the tree.

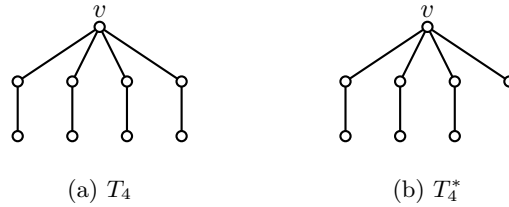


Figure 6.6: The subdivided star  $T_4 = S(K_{1,4})$

Our main result is to prove that, not only for trees but also for graphs with no 4-cycles, among all graphs of given maximum degree, the subdivided stars described above require the largest fraction of their vertex set in any optimal OTD-code. More precisely, we prove the following statement.

**Theorem 6.5.** *For  $\Delta \geq 3$  a fixed integer, if  $G$  is an open-twin-free connected graph of order  $n \geq 5$  that contains no 4-cycles and satisfies  $\Delta(G) \leq \Delta$ , then*

$$\gamma^{\text{OTD}}(G) \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n,$$

*except in one exceptional case when  $G = T_\Delta$ , in which case  $\gamma^{\text{OTD}}(G) = \left( \frac{2\Delta}{2\Delta+1} \right) n$ .*

Moreover, we show that the above bound is tight for every value of  $\Delta \geq 3$ . Furthermore, when  $\Delta = 3$ , we give a construction that provides infinitely many connected graphs for which the bound is tight.

### 6.1.3.2 Trees

In this section, we prove Theorem 6.5 for trees. Towards this goal, we first define some special families of trees that will be essential in the proof.

#### A special family $\mathcal{T}$ of trees

In this section, we define a special family of trees that we will often need when proving our main result. In order to define our special family of trees, we introduce some additional notations. Let  $r$  be a specified vertex in a tree  $T$ . We define next several types of attachments at the vertex  $r$  that we use to build larger trees. In all cases, we call the vertex of the attachment that is joined to  $r$  the *link vertex* of the attachment.

- For  $i \in [4]$ , an *attachment of Type- $i$  at  $r$*  is an operation that adds a path  $P_i$  to  $T$  and joins one of its ends to  $r$ .
- An *attachment of Type-5 at  $r$*  is an operation that adds a path  $P_4$  to  $T$  and joins one of its support vertices to  $r$ .
- An *attachment of Type-6 at  $r$*  is an operation that adds a star  $K_{1,3}$  with one edge subdivided and joins one of its leaves that is adjacent to the vertex of degree 3 to  $r$ .

Let  $\mathbb{N}$  be the set of all non-negative integers and let

$$\begin{aligned} \mathbb{N}_*^6 = \{ \mathbf{k} = (k_1, k_2, k_3, k_4, k_5, k_6) \in \mathbb{N}^6 : k_1 \in \{0, 1\} \text{ and} \\ \mathbf{k} \neq (1, 0, 0, 0, 0, 0), (0, 1, 0, 0, 0, 0), (0, 0, 1, 0, 0, 0), (1, 1, 0, 0, 0, 0) \}. \end{aligned} \quad (6.3)$$

From now on, let  $\mathbf{k}$  denote a vector in  $\mathbb{N}_*^6$ , let  $k_i$  denote the  $i$ th coordinate of  $\mathbf{k}$  and let  $k = \sum_{i=1}^6 k_i$ . We define  $T(r; \mathbf{k})$  to be the tree obtained from a trivial tree  $K_1$  whose vertex is named  $r$  by applying  $k_i$  attachments of Type- $i$  at vertex  $r$  for each  $i \in [6]$ . Notice that  $\deg_T(r) = k$ . We now define the set  $\mathcal{T} = \{T(r; \mathbf{k}) : \mathbf{k} \in \mathbb{N}_*^6\}$  to be our special family of trees. We note that the forbidden configurations of the vector  $\mathbf{k}$  in the definition of  $\mathbb{N}_*^6$  exclude the trees  $P_2$ ,  $P_3$  and  $P_4$  from the family  $\mathcal{T}$ , as in our main result, we only consider graphs of order at least 5.

The tree  $T(r; 1, 3, 2, 3, 2, 2)$ , for example, is illustrated in Figure 6.7(a). For  $k \geq 2$ , the subdivided star  $T_k$  is isomorphic to the tree  $T(r; 0, k, 0, 0, 0, 0) \in \mathcal{T}$  and also to the tree  $T(r; 1, 0, 0, 0, 1, 0) \in \mathcal{T}$  for  $k = 2$  (see Figure 6.7(c)). For  $k \geq 3$ , the reduced subdivided star  $T_k^*$  is isomorphic to the tree  $T(r; 1, k-1, 0, 0, 0, 0) \in \mathcal{T}$  and also to the tree  $T(r; 1, 0, 0, 0, 1, 0)$  for  $k = 2$  (see Figure 6.7(b)).

**Definition 6.1.** We define the *canonical set*  $C$  of a tree  $T = T(r; \mathbf{k}) \in \mathcal{T}$  as follows. For  $\mathbf{k} \notin \{(1, 0, 1, 0, 0, 0), (1, 0, 0, 0, 1, 0)\}$ , that is, when  $T$  is neither  $T_2$  nor  $T_3^*$  with  $r$  being a degree 2 support vertex of  $T$  (as illustrated in Figures 6.7(b) and (c), respectively), we define the canonical set  $C$  of  $T$  as follows.

- Initially, we add the vertex  $r$  to the set  $C$ .
- If  $k_1 = 1$  and  $k_2 \geq 1$ , then we add to  $C$  all vertices that belong to attachments of Type-2 at  $r$ , but we do not add to  $C$  the vertex that belongs to the attachment of Type-1 at  $r$ .
- If  $k_1 = 0$  and  $k_2 \geq 1$ , then we add to  $C$  all vertices that belong to attachments of Type-2 at  $r$ , except for one leaf at distance 2 from  $r$  in  $T$  in exactly one attachment of Type-2.
- If  $k_3 \geq 1$ , then we add to  $C$  all vertices that belong to attachments of Type-3 at  $r$  that are not leaves at distance 3 from  $r$  in  $T$ .
- If  $k_4 \geq 1$ , then we add to  $C$  all vertices that belong to attachments of Type-4 at  $r$  that are not leaves at distance 4 from  $r$  in  $T$ .
- If  $k_5 \geq 1$ , then we add to  $C$  all vertices of attachments of Type-5 at  $r$  except for the leaves at distance 2 from  $r$  in  $T$ .
- If  $k_6 \geq 1$ , then we add to  $C$  all vertices of attachments of Type-6 at  $r$  except for the leaves at distance 3 from  $r$  in  $T$ .

For  $\mathbf{k} \in \{(1, 0, 1, 0, 0, 0), (1, 0, 0, 0, 1, 0)\}$ , that is, when  $T$  is either  $T_2$  or  $T_3^*$  with  $r$  being a degree 2 support vertex of  $T$  (as in Figures 6.7(b) and (c), respectively), we define the canonical set  $C$  of  $T$  as follows.

- For  $\mathbf{k} = (1, 0, 1, 0, 0, 0)$ , add to  $C$  all vertices of  $T$  except for the leaf of  $T$  at distance 3 of  $r$ .
- For  $\mathbf{k} = (1, 0, 1, 0, 0, 0)$ , add to  $C$  all vertices of  $T$  except for the leaf of  $T$  at distance 2 of  $r$ .

For the trees  $T(r; 1, 3, 2, 3, 2, 2)$ ,  $T(r; 1, 0, 1, 0, 0, 0)$  and  $T(r; 1, 0, 0, 0, 1, 0)$ , for example, the shaded vertices in Figures 6.7(a), 6.7(b) and 6.7(c), respectively, indicate the canonical set  $C$  of the tree. Notice that the two trees  $T(r; \mathbf{k})$  for  $\mathbf{k} \in \{(0, 2, 0, 0, 0, 0), (1, 0, 1, 0, 0, 0)\}$  (both isomorphic to  $T_2$ ) have the same canonical sets as prescribed in Definition 6.1. This property also holds for the trees  $T(r; \mathbf{k})$  for  $\mathbf{k} \in \{(1, 2, 0, 0, 0, 0), (1, 0, 0, 0, 1, 0)\}$  (both isomorphic to  $T_3^*$ ).

**Observation 6.1.** *The canonical set of any tree  $T = T(r; \mathbf{k}) \in \mathcal{T}$  is also an OTD-code of  $T$ .*

We next determine the OTD-numbers of (reduced) subdivided stars  $T_k$  and  $T_k^*$ .

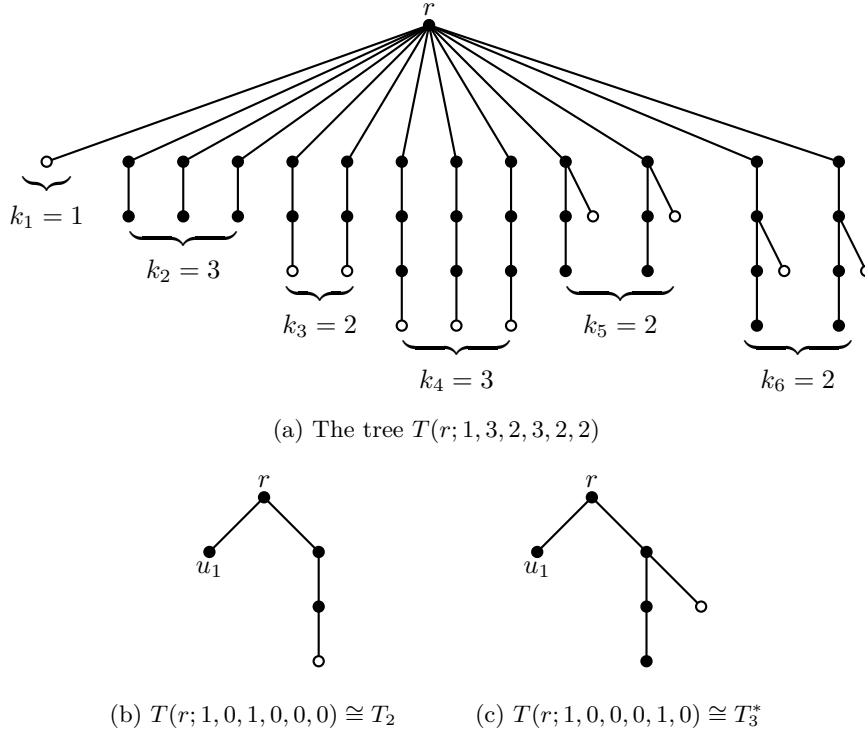
**Proposition 6.3.** *For  $k \geq 2$  a fixed integer and a tree  $T$  of order  $n$ , the following properties hold.*

- (a) *If  $T = T_k$ , then  $\gamma^{\text{OTD}}(T) = \left(\frac{2k}{2k+1}\right)n$ .*
- (b) *If  $k \geq 3$  and  $T = T_k^*$ , then  $\gamma^{\text{OTD}}(T) = \left(\frac{2k-1}{2k}\right)n$ .*

*Moreover, in both cases, the canonical set of  $T$  is an optimal OTD-code of  $T$ .*

*Proof.* (a) Let  $T = T_k$  have order  $n$  where  $k \geq 2$ , and let  $v$  be the central vertex of  $T$  (of degree  $k$ ). Thus,  $n = 2k + 1$ . Let  $S$  be an OTD-code of  $T$ . Since  $S$  is a TD-set of  $T$ , in order to totally dominate all the leaves in  $T$  the set  $S$  contains all support vertices of  $T$ . Since  $S$  is a separating open code, in order to identify the support vertices of  $T$  the set  $S$  contains all, except possibly one leaf. If  $S$  contains all leaves of  $T$ , then  $|S| \geq 2k$ . Suppose that  $S$  does not contain all leaves of  $T$ . By our earlier observations, in this case there is exactly one leaf,  $u$  say, that does not belong to  $S$ . Let  $w$  be the support vertex adjacent to  $u$ . Thus,  $N_T(w) = \{u, v\}$ . In order to totally dominate the vertex  $w$ , we infer that  $v \in S$ , implying once again that  $|S| \geq 2k$ . Since  $S$  is an arbitrary OTD-code of  $T$ , this implies that  $\gamma^{\text{OTD}}(T) \geq 2k$ . Now, assuming  $T \cong T(r; 0, k, 0, 0, 0, 0) \in \mathcal{T}$ , let  $C$  be the canonical set of the tree  $T$  as in Definition 6.1, that is,  $C$  is the set of all vertices of  $T$  except for one leaf of  $T$ . Then, by Observation 6.1, the set  $C$  is an OTD-code and it satisfies  $\gamma^{\text{OTD}}(T) \leq |C| = 2k$ . Consequently,  $\gamma^{\text{OTD}}(T) = 2k = \left(\frac{2k}{2k+1}\right)n$ .

(b) Let  $T = T_k^*$  have order  $n$  where  $k \geq 3$ , and let  $v$  be the central vertex of  $T$  (of degree  $k$ ) and let  $x$  denote the leaf neighbor of  $v$ . Thus,  $n = 2k$ . Let  $S$  be an OTD-code of  $T$ . Since  $S$  is a

Figure 6.7: The canonical sets of a general tree and the trees  $T_2$  and  $T_3^*$  in  $\mathcal{T}$ .

TD-set of  $T$ , the set  $S$  contains all support vertices of  $T$ . Since  $S$  is an OTD-code of  $T$ , in order to identify the leaf  $x$  from all other neighbors of  $v$ , the set  $S$  contains all leaves of  $T$  at distance 2 from  $v$ . Thus,  $S$  contains all vertices of  $T$ , except possibly for the leaf  $x$  of  $T$ . Thus,  $|S| \geq 2k - 1$ . Since  $S$  is an arbitrary OTD-code of  $T$ , this implies that  $\gamma^{\text{OTD}}(T) \geq 2k - 1$ . On the other hand, assuming  $T \cong T(r; 1, k - 1, 0, 0, 0, 0)$ , let  $C$  be the canonical set of  $T$  as in Definition 6.1, that is,  $C$  consists of all vertices of  $T$  except for the leaf neighbor  $x$  of  $v$ . Again, by Observation 6.1, the set  $C$  is an OTD-code of  $T$ . Moreover, we have  $\gamma^{\text{OTD}}(T) \leq |C| = 2k - 1$ . Consequently,  $\gamma^{\text{OTD}}(T) = 2k - 1 = \binom{2k-1}{2k} n$ .  $\square$

We next prove that the bound of Theorem 6.5 holds for the trees in  $\mathcal{T}$ .

**Proposition 6.4.** *Let  $\Delta \geq 3$  be a fixed integer and let  $T = T(r; \mathbf{k}) \in \mathcal{T}$  be a tree of order  $n \geq 5$ , maximum degree  $\Delta(T) \leq \Delta$  such that  $\deg_T(r) = k \geq 2$ . If  $T$  is not isomorphic to the subdivided star  $T_\Delta$ , then*

$$\gamma^{\text{OTD}}(T) \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n.$$

*Proof.* By supposition,  $\Delta \geq \Delta(T) \geq \deg_T(r) = k$ . Therefore, if  $T \cong T_k$  for some  $k \leq \Delta - 1$ , then the result follows from Proposition 6.3. Notice that  $T$  cannot be isomorphic to the reduced subdivided star  $T_2^* \cong P_4$ , since  $T \in \mathcal{T}$  and the latter does not contain the path  $P_4$ . On the other hand, if  $T$  is isomorphic to the reduced subdivided star  $T_k^*$  for  $k \geq 3$ , then again the desired upper bound follows from Proposition 6.3. Hence, we may assume that the tree  $T$  is neither isomorphic to  $T_k$  nor to  $T_k^*$  for all  $2 \leq k \leq \Delta$ . Therefore, in particular,  $T$  is neither the tree  $T(r; 1, 0, 1, 0, 0, 0)$  as in Figure 6.7(b) nor the tree  $T(r; 1, 0, 0, 0, 1, 0)$  as in Figure 6.7(c).

Let  $C$  be the canonical set of  $T = T(r; \mathbf{k})$  as in Definition 6.1. Let  $A$  be the set of vertices of  $T$  consisting of the vertex  $r$  and all vertices that belong to attachments of Type-1 and of Type-2 at  $r$ , and let  $B = V(T) \setminus A$ . Further, let  $a = |A|$  and  $b = |B|$ , and so  $n = a + b$ . Since  $T \not\cong T_k$  and  $T \not\cong T_k^*$ ,



we must have  $k_1 + k_2 \leq k - 1$ . We note that  $1 \leq a \leq 2k - 1$ . Let  $C_1 = C \cap A$ . Since  $\mathbf{k} \neq (1, 0, 1, 0, 0, 0)$  and  $(1, 0, 0, 0, 1, 0)$ , by construction of the canonical set  $C$ , we have  $|C_1| = a - 1 \leq \left(\frac{2k-2}{2k-1}\right)a$ . Further, let  $C_2 = C \cap B$ , and so  $|C_2| \leq \frac{4}{5}b$ . Since  $C$  is an OTD-code of  $T$  by Observation 6.1 and  $\Delta \geq 3$ , we infer that

$$\gamma^{\text{OTD}}(T) \leq |C| = |C_1| + |C_2| \leq \left(\frac{2k-2}{2k-1}\right)a + \frac{4}{5}b < \left(\frac{2\Delta-1}{2\Delta}\right)n.$$

□

### Theorem 6.5 for trees

In this section, we prove the following key result which we will need when proving our main result, namely Theorem 6.5, for graphs that contain no 4-cycles.

**Theorem 6.6.** *For  $\Delta \geq 3$  a fixed integer, if  $T$  is an open-twin-free tree of order  $n \geq 5$  that satisfies  $\Delta(T) \leq \Delta$ , then*

$$\gamma^{\text{OTD}}(T) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n,$$

*except in one exceptional case when  $T = T_\Delta$ , in which case  $\gamma^{\text{OTD}}(T) = \left(\frac{2\Delta}{2\Delta+1}\right)n$ .*

*Proof.* Let  $\Delta \geq 3$  be a fixed integer. We proceed by induction on the order  $n \geq 5$  of a tree  $T$  that is open-twin-free and satisfies  $\Delta(T) \leq \Delta$ . Since  $T$  has order  $n \geq 5$  and is twin-free, we note that  $\text{diam}(T) \geq 4$  and  $T$  has no strong support vertices, that is, every support has exactly one leaf neighbor. We proceed further with a series of claims.

■ **Claim 2.** If  $\text{diam}(T) = 4$ , then one of the following holds.

- (a)  $T = T_\Delta$  and  $\gamma^{\text{OTD}}(T) = \left(\frac{2\Delta}{2\Delta+1}\right)n$ .
- (b)  $T \neq T_\Delta$  and  $\gamma^{\text{OTD}}(T) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n$ .

*Proof of claim.* Suppose that  $\text{diam}(T) = 4$ . Since  $T$  has no strong support vertex, either  $T = T_k$  for some  $k$  where  $2 \leq k \leq \Delta$  or  $T = T_k^*$  for some  $k$  where  $3 \leq k \leq \Delta - 1$ . If  $T = T_\Delta$ , then by Proposition 6.3(a), we have  $\gamma^{\text{OTD}}(T) = \left(\frac{2\Delta}{2\Delta+1}\right)n$ , and so statement (a) of the claim holds. If  $T = T_k$  for some  $k$  where  $2 \leq k \leq \Delta - 1$ , then by Proposition 6.3(a),

$$\gamma^{\text{OTD}}(T) = \left(\frac{2k}{2k+1}\right)n \leq \left(\frac{2\Delta-2}{2\Delta-1}\right)n < \left(\frac{2\Delta-1}{2\Delta}\right)n.$$

If  $T = T_k^*$  for some  $k$  where  $3 \leq k \leq \Delta - 1$ , then by Proposition 6.3(b),

$$\gamma^{\text{OTD}}(T) = \left(\frac{2k-1}{2k}\right)n \leq \left(\frac{2\Delta-1}{2\Delta}\right)n,$$

and so statement (b) of the claim holds. ■

By Claim 2, we may assume that  $\text{diam}(T) \geq 5$ , for otherwise the desired result holds. In what follows, let  $e = v_1v_2$  be an edge of  $T$  and let  $F_i$  be the component of  $T - e$  that contains the vertex  $v_i$  for  $i \in [2]$ . Further, let  $F_i$  have order  $n_i$  where  $i \in [2]$ , and so  $n = n_1 + n_2$ . We note that  $T$  is obtained from the trees  $F_1$  and  $F_2$  by adding back the edge  $e$ .

■ **Claim 3.** If  $F_1 = T_k$  and  $v_1$  is the central vertex of degree  $k$  in  $F_1$  for some  $k$  where  $2 \leq k \leq \Delta - 1$ , then

$$\gamma^{\text{OTD}}(T) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n.$$

*Proof of claim.* Suppose that  $F_1 = T_k$  and  $v_1$  is the central vertex of degree  $k$  in  $F_1$  for some  $k$  where  $2 \leq k \leq \Delta - 1$ . Without loss of generality, we may assume that  $F_1 = T(v_1; 0, k, 0, 0, 0, 0) \in \mathcal{T}$ . Let  $S_1$  denote the canonical set of the tree  $F_1$ . Thus,  $S_1$  contains all vertices of  $F_1$ , except for exactly one leaf,  $u_1$  say. In particular, we note that  $S_1$  contains the vertex  $v_1$  and all support vertices of  $F_1$ . We first consider the case when  $n_2 \leq 4$ . Since  $T \neq T_\Delta$ , we have  $n_2 \neq 2$ . Moreover, for  $n_2 \in \{1, 3, 4\}$ , the tree  $T$  is isomorphic to some tree in the special family  $\mathcal{T}$ . Therefore, in the case that  $n_2 \leq 4$ , we are done by Proposition 6.4. Hence, we may assume that  $n_2 \geq 5$ .

Let us now assume that  $F_2$  is open-twin-free. Suppose that  $F_2 = T_\Delta$ . In this case, either  $v_2$  is a support vertex in  $F_2$  (as illustrated in Figure 6.8(a)) or  $v_2$  is a leaf in  $F_2$  (as illustrated in Figure 6.8(b)). In both cases, let  $S_1 = V(F_1) \setminus \{u_1\}$ . If  $v_2$  is a support vertex in  $F_2$ , then let  $S_2$  consist of all vertices of  $F_2$  except for exactly two leaves, one of which is the leaf neighbor of  $v_2$  in  $F_2$ . If  $v_2$  is a leaf in  $F_2$ , then let  $S_2$  consist of all vertices of  $F_2$  except for exactly one leaf which is different from the vertex  $v_2$ . Let  $S = S_1 \cup S_2$ . In the first case when  $v_2$  is a support vertex in  $F_2$ , the resulting set  $S$  is illustrated by the shaded vertices in Figure 6.8(a). In the second case when  $v_2$  is a leaf in  $F_2$ , the resulting set  $S$  is illustrated by the shaded vertices in Figure 6.8(b). In both cases, it can be verified that the set  $S$  is an OTD-code of  $T$ . Thus, since  $|S| = |S_1| + |S_2| \leq 2k + 2\Delta$  and  $n = (2k + 1) + (2\Delta + 1)$ , and since  $k \leq \Delta - 1$ , we infer that

$$\gamma^{\text{OTD}}(T) \leq |S| \leq \left( \frac{2k + 2\Delta}{2k + 2\Delta + 2} \right) n = \left( \frac{k + \Delta}{k + \Delta + 1} \right) n \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n.$$

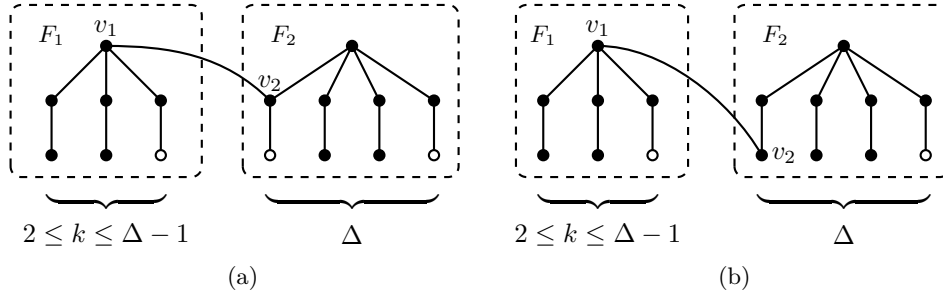


Figure 6.8: Possible trees in the proof of Claim 3

Hence we may assume that  $F_2 \neq T_\Delta$ . Applying the inductive hypothesis to the open-twin-free tree  $F_2$  of order  $n_2 \geq 5$ , there exists an OTD-code,  $S_2$  say, of  $F_2$  such that  $\gamma^{\text{OTD}}(F_2) = |S_2| \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n_2$ . Let  $S = S_1 \cup S_2$ . Since the vertex  $v_2$  in  $F_2$  is totally dominated by some other vertex of  $F_2$  in  $S_2$ , the set  $S$  is an OTD-code of the tree  $T$ . By Proposition 6.4, we have  $|S_1| \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n_1$ , and so

$$\gamma^{\text{OTD}}(T) \leq |S| = |S_1| + |S_2| \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n_1 + \left( \frac{2\Delta - 1}{2\Delta} \right) n_2 = \left( \frac{2\Delta - 1}{2\Delta} \right) n.$$

Let us now assume that  $F_2$  contains open twins. Since the tree  $T$  contains no open twins, we infer that  $v_2$  is a leaf in  $F_2$  and is an open twin in  $F_2$  with some other leaf, say  $u_2$ , in  $F_2$ . Let  $v$  be the common neighbor of  $v_2$  and  $u_2$  in  $F_2$ . Let  $T' = F_2 - v_2$ , and let  $T'$  have order  $n'$ , and so  $n' = n_2 - 1 \geq 4$ . Since  $T$  is open-twin-free, so too is the tree  $T'$ .

Suppose that  $n' = 4$ , and so  $n_2 = 5$ . In this case, the tree  $T \cong T(v_1; 0, k, 0, 0, 0, 1) \in \mathcal{T}$  and therefore, we are done by Proposition 6.4. Hence we may assume that  $n' \geq 5$ . Suppose that  $T' = T_\Delta$ . In this case, the vertex  $v$  is a support vertex in the tree  $T'$  and the tree  $T$  is illustrated in Figure 6.9. Let  $u_2$  denote the leaf neighbor of the vertex  $v$ . We note that  $n_1 = 2k + 1$  and  $n_2 = 2\Delta + 2$ , and so  $n = 2k + 2\Delta + 3$ . We now let  $S_2$  consist of all vertices in  $T'$ , except for the leaf  $u_2$ . Thus,  $|S_2| = 2\Delta$ . Further, we let  $S = S_1 \cup S_2$ , and so the set  $S$  is indicated by the shaded vertices in Figure 6.9. It can be verified that the set  $S$  is an OTD-code of the tree, and so

$$\gamma^{\text{OTD}}(T) \leq |S| = 2k + 2\Delta = \left( \frac{2k + 2\Delta}{2k + 2\Delta + 3} \right) n \leq \left( \frac{4\Delta - 2}{4\Delta + 1} \right) n < \left( \frac{2\Delta - 1}{2\Delta} \right) n.$$

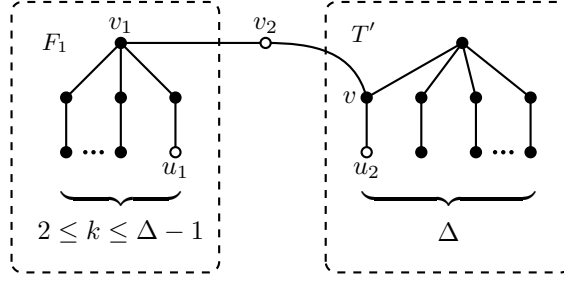


Figure 6.9: A possible tree in the proof of Claim 3

Hence, we may assume that  $T' \neq T_\Delta$ . Recall that  $n' \geq 5$ . Applying the inductive hypothesis to the open-twin-free tree  $T'$  of order  $n' \geq 5$ , there exists an OTD-code,  $S'$  say, of  $T'$  such that  $\gamma^{\text{OTD}}(T') = |S'| \leq \left(\frac{2\Delta-1}{2\Delta}\right)n'$ . Now let  $S = S_1 \cup S'$ . Since  $u_2$  is a leaf in  $T'$ , the OTD-code  $S'$  of  $T'$  contains the support vertex  $v$ , which has two neighbors in  $S$ . Therefore, the set  $S$  is an OTD-code of the tree  $T$ , and so

$$\gamma^{\text{OTD}}(T) \leq |S| = |S_1| + |S'| \leq \left(\frac{2\Delta-1}{2\Delta}\right)n_1 + \left(\frac{2\Delta-1}{2\Delta}\right)n' < \left(\frac{2\Delta-1}{2\Delta}\right)n.$$

This completes the proof of Claim 3. ■

By Claim 3, we may assume that the removal of any edge from  $T$  does not produce a component  $T'$  isomorphic to  $T_k$ , where  $2 \leq k \leq \Delta - 1$  and where the central vertex of the subdivided star  $T'$  is incident with the deleted edge, for otherwise the desired result follows. As a consequence of Claim 3, we have the following property of the tree  $T$ .

■ **Claim 4.** If  $T$  contains an edge whose removal produces a component isomorphic to  $T_\Delta$ , then

$$\gamma^{\text{OTD}}(T) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n.$$

*Proof of claim.* Suppose that  $T$  contains an edge  $f = xy$  whose removal produces a component,  $T'$  say, isomorphic to  $T_\Delta$ . Renaming  $x$  and  $y$  if necessary, we may assume that  $x \in V(T')$ . Let  $v$  be the central vertex of the subdivided star in  $T'$ , and so  $v$  has degree  $\Delta$  in  $T'$  (and in the original tree  $T$ ). If  $x$  is a support vertex of  $T'$ , then we infer from Claim 3 with  $v_1 = v$  and  $v_2 = x$  that  $\gamma^{\text{OTD}}(T) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n$ . If  $x$  is a leaf of  $T'$ , then we let  $w$  denote the common neighbor of  $v$  and  $x$ , and we infer from Claim 3 with  $v_1 = v$  and  $v_2 = w$  that  $\gamma^{\text{OTD}}(T) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n$ . ■

By Claim 4, we may assume that the removal of any edge from  $T$  does not produce a component isomorphic to  $T_\Delta$ , for otherwise the desired result follows.

For the rest of the proof, let  $v_0v_1v_2 \dots v_d$  to be a fixed longest path in  $T$ . Then  $v_0$  and  $v_d$  are leaves in  $T$ . Let us also assume from here on that the tree  $T$  is rooted at the leaf  $v_d$ . Since  $\text{diam}(T) \geq 5$  by assumption, we must have  $d \geq 5$ . This implies that, for all  $i \in [5]$ , the vertex  $v_i$  is the ancestor of  $v_{i-1}$  in  $T$ . If  $v$  is a vertex in  $T$ , then we denote by  $T_v$  the maximal subtree rooted at  $v$ , and so  $T_v$  consists of  $v$  and all descendants of the vertex  $v$ . Since  $T$  has no strong support vertex, we note that  $\deg_T(v_1) = 2$ .

■ **Claim 5.** If either  $\deg_T(v_2) \geq 4$  or  $\deg_T(v_i) \geq 3$  for  $i \in \{3, 4\}$ , then

$$\gamma^{\text{OTD}}(T) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n.$$

*Proof of claim.* We first assume that  $\deg_T(v_2) \geq 4$ . Then, the maximal subtree  $T_{v_2}$  of  $T$  rooted at  $v_2$  is isomorphic to a tree in the special family  $\mathcal{T}$ . By Claim 3, we may assume that  $T_{v_2} \not\cong T_k$  for any  $k \leq \Delta - 1$ . Hence,  $T_{v_2}$  must be isomorphic to the reduced subdivided tree  $T_k^*$ . Let  $T_{v_2}$  be on  $n_1$  vertices and let  $T'$  be the component of  $T - v_2v_3$  that contains the vertex  $v_3$ , and let  $T'$  have order  $n'$ . If  $n' \leq 4$ , then  $T$  is isomorphic to a tree in the special family  $\mathcal{T}$  and thus, we are done by Proposition 6.4. Let us therefore assume that  $n' \geq 5$ . By Claim 4, we also assume that  $T'$  is not isomorphic to  $T_\Delta$ .

First, we assume that  $T'$  is open-twin-free. Therefore, applying the inductive hypothesis to the open-twin-free tree  $T'$ , there exists an OTD-code,  $S'$  say, of  $T'$  such that  $\gamma^{\text{OTD}}(T') = |S'| \leq \left(\frac{2\Delta-1}{2\Delta}\right) n'$ . Let  $C$  be the canonical set of  $T_{v_2}$ . Then, by Observation 6.1, the set  $C$  is an OTD-code of the tree  $T_{v_2}$  and by Proposition 6.3, we have  $|C| \leq \left(\frac{2\Delta-1}{2\Delta}\right) n_1$ . Let  $S = C \cup S'$ . Then it can be verified that the set  $S$  is an OTD-code of the tree  $T$ , and so

$$\begin{aligned} \gamma^{\text{OTD}}(T) \leq |S| &= |C| + |S'| \\ &\leq \left(\frac{2\Delta-1}{2\Delta}\right) n_1 + \left(\frac{2\Delta-1}{2\Delta}\right) n' \\ &= \left(\frac{2\Delta-1}{2\Delta}\right) n. \end{aligned}$$

We now suppose that  $T'$  contains open twins. Since the tree  $T$  contains no open twins, we infer that  $v_3$  is a leaf in  $T'$  and is an open twin in  $T'$  with some other leaf, say  $u_3$ , in  $T'$ . We note that  $v_4$  is the common neighbor of  $v_3$  and  $u_3$ . Let  $T'' = T' - v_3$ , and let  $T''$  have order  $n''$ , and so  $n'' = n' - 1 \geq 4$ . If  $n'' = 4$ , since  $v_3$  and  $u_3$  are open twins in  $T''$ , it implies that  $T \cong T(v_2; 1, k-2, 0, 0, 0, 1) \in \mathcal{T}$  for some  $k = \deg_T(v_2) - 1 \leq \Delta - 1$ . Hence, in this case, we are again done by Proposition 6.4. Let us therefore assume that  $n'' \geq 5$ . Since  $T$  is open-twin-free, so too is the tree  $T''$ . By Claim 4, we infer that  $T'' \not\cong T_\Delta$ . Applying the inductive hypothesis to the open-twin-free tree  $T''$  of order  $n'' \geq 5$ , there exists an OTD-code,  $S''$  say, of  $T''$  such that  $\gamma^{\text{OTD}}(T'') = |S''| \leq \left(\frac{2\Delta-1}{2\Delta}\right) n''$ . Let  $S = C \cup S''$ . We note that since  $u_3$  is a leaf, the vertex  $v_4 \in S''$  and so the vertex  $v_3$  has two neighbors in the set  $S$  and thus is identified by  $S$ . Hence, the set  $S = C \cup S''$  is an OTD-code of  $T$ , and so

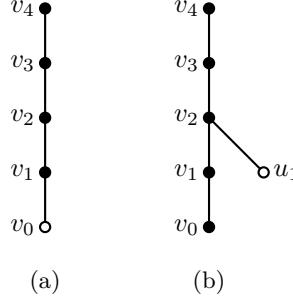
$$\begin{aligned} \gamma^{\text{OTD}}(T) \leq |S| &= |C| + |S''| \\ &\leq \left(\frac{2\Delta-1}{2\Delta}\right) n_1 + \left(\frac{2\Delta-1}{2\Delta}\right) n'' \\ &< \left(\frac{2\Delta-1}{2\Delta}\right) (n_1 + 1) + \left(\frac{2\Delta-1}{2\Delta}\right) n'' \\ &= \left(\frac{2\Delta-1}{2\Delta}\right) (n - n'') + \left(\frac{2\Delta-1}{2\Delta}\right) n'' \\ &= \left(\frac{2\Delta-1}{2\Delta}\right) n. \end{aligned}$$

This proves the result in the case that  $\deg_T(v_2) \geq 4$ . The case for  $\deg_T(v_3) \geq 3$  (respectively,  $\deg_T(v_4) \geq 3$ ) follows by exactly the same arguments as for the case when  $\deg_T(v_2) \geq 4$  by simply taking  $T'$  to be the component of  $T - v_3v_4$  (respectively,  $T - v_4v_5$ ) containing the vertex  $v_4$  (respectively,  $v_5$ ). This proves the claim.  $\blacksquare$

Hence, by Claim 5, we can assume from here on that  $2 \leq \deg_T(v_2) \leq 3$  and  $\deg_T(v_i) = 2$  for all  $i \in \{3, 4\}$ , or else, the desired result is achieved. We now look at the maximal subtree  $T_{v_4}$  of  $T$  rooted at  $v_4$ . By our earlier assumptions, either  $T_{v_4}$  is a path  $P_5$  with  $v_4$  as one of the ends of the path as illustrated in Figure 6.10(a), or  $T_{v_4}$  is a reduced subdivided star  $T_3^*$  as illustrated in Figure 6.10(b) with  $v_2$  as the central vertex of the reduced subdivided star. In the latter case, we let  $u_1$  be the leaf neighbor of  $v_2$ .

If  $T_{v_4}$  is a path  $P_5$  with  $v_4$  as one of the ends of the path, then we let  $S_1 = \{v_1, v_2, v_3, v_4\}$ . In this case, the set  $S_1$ , indicated by the shaded vertices in Figure 6.10(a), is an OTD-code of  $T_{v_4}$ . If  $T_{v_4}$  is a reduced subdivided star  $T_3^*$ , then we let  $S_1 = \{v_0, v_1, v_2, v_3, v_4\}$ . In this case, the set  $S_1$ , indicated by the shaded vertices in Figure 6.10(b), is an OTD-code of  $T_{v_4}$ .

Let  $T'$  be the component of  $T - v_4v_5$  that contains the vertex  $v_5$ , and let  $T'$  have order  $n'$ . If  $n' \leq 4$ , then the tree  $T$  is determined and is isomorphic to a tree in  $\mathcal{T}$ , and hence, we infer the desired upper bound by Proposition 6.4. We may therefore assume that  $n' \geq 5$ .

Figure 6.10: Two possible subtrees in  $T$ 

Suppose now that  $T'$  is open-twin-free. By our earlier assumptions, we infer that  $T' \neq T_\Delta$ . Applying the inductive hypothesis to the open-twin-free tree  $T'$ , there exists an OTD-code,  $S'$  say, of  $T'$  such that  $\gamma^{\text{OTD}}(T') = |S'| \leq \left(\frac{2\Delta-1}{2\Delta}\right) n'$ . Let  $S = S_1 \cup S'$ . It can be verified that the set  $S$  is an OTD-code of the tree  $T$ . If  $T'$  is a path  $P_5$ , then  $n_1 = 5$  and  $|S_1| = 4 = \frac{4}{5}n_1 = \frac{4}{5}(n - n')$ . If  $T'$  is a reduced subdivided star  $T_3^*$ , then  $n_1 = 6$  and  $|S_1| = 5 = \frac{5}{6}n_1 = \frac{5}{6}(n - n')$ . In both cases,  $|S_1| \leq \frac{5}{6}(n - n')$ . Therefore,

$$\begin{aligned} \gamma^{\text{OTD}}(T) \leq |S| &= |S_1| + |S'| \\ &\leq \frac{5}{6}(n - n') + \left(\frac{2\Delta-1}{2\Delta}\right) n' \\ &\leq \left(\frac{2\Delta-1}{2\Delta}\right) (n - n') + \left(\frac{2\Delta-1}{2\Delta}\right) n' \\ &= \left(\frac{2\Delta-1}{2\Delta}\right) n. \end{aligned}$$

Suppose now that  $T'$  contains open twins. Since the tree  $T$  contains no open twins, we infer that  $v_5$  is a leaf in  $T'$  and is an open twin in  $T'$  with some other leaf, say  $u_5$ , in  $T'$ . We note that  $v_6$  is the common neighbor of  $v_5$  and  $u_5$ . Let  $T'' = T' - v_5$ , and let  $T''$  have order  $n''$ , and so  $n'' = n' - 1 \geq 4$ . If  $n'' = 4$ , then  $T$  is isomorphic to  $T(v_4; 0, 0, 0, 1, 0, 1)$  if  $T' \cong P_5$  and to  $T(v_4; 0, 0, 0, 0, 0, 2)$  if  $T' \cong T_3^*$ . In both cases, the result follows by Proposition 6.4. Therefore, let us assume that  $n'' \geq 5$ . Since  $T$  is open-twin-free, so too is the tree  $T''$ . By our earlier observations, we infer that  $T'' \neq T_\Delta$ . Applying the inductive hypothesis to the open-twin-free tree  $T''$  of order  $n'' \geq 5$ , there exists an OTD-code,  $S''$  say, of  $T''$  such that  $\gamma^{\text{OTD}}(T'') = |S''| \leq \left(\frac{2\Delta-1}{2\Delta}\right) n''$ . Let  $S = S_1 \cup S''$ . We note that since  $u_5$  is a leaf, the vertex  $v_6 \in S''$ , and so  $v_5$  is identified by the set  $S$ . Therefore, the set  $S$  is an OTD-code of the tree  $T$ . If  $T_{v_4}$  is a path  $P_5$ , then  $n_1 = 5$  and  $|S_1| = 4 = \frac{2}{3}(n_1 + 1) = \frac{2}{3}(n - n'')$ . If  $T_{v_4}$  is a reduced subdivided star  $T_3^*$ , then  $n_1 = 6$  and  $|S_1| = 5 = \frac{5}{7}(n_1 + 1) = \frac{5}{7}(n - n'')$ . In both cases,  $|S_1| \leq \frac{5}{7}(n - n'')$ . Therefore,

$$\begin{aligned} \gamma^{\text{OTD}}(T) \leq |S| &= |S_1| + |S''| \\ &\leq \frac{5}{7}(n - n'') + \left(\frac{2\Delta-1}{2\Delta}\right) n'' \\ &< \left(\frac{2\Delta-1}{2\Delta}\right) (n - n'') + \left(\frac{2\Delta-1}{2\Delta}\right) n'' \\ &= \left(\frac{2\Delta-1}{2\Delta}\right) n. \end{aligned}$$

This completes the proof of Theorem 6.6.  $\square$

### 6.1.3.3 Proof of Theorem 6.5

In this section, we present a proof of our main result, namely Theorem 6.5. We first present the following preliminary result.

**Proposition 6.5.** *For  $\Delta \geq 3$  a fixed integer, if  $G$  is a graph of order  $n$  that is obtained from a subdivided star  $T_k$ , where  $2 \leq k \leq \Delta$ , by adding an edge  $e$  to  $T_k$  in such a way that  $G = T_k + e$*

contains no 4-cycle and satisfies  $\Delta(G) \leq \Delta$ , then

$$\gamma^{\text{OTD}}(G) \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n.$$

*Proof.* It can be verified that the graph  $G$  in the statement of the proposition must be isomorphic to exactly one of the three graphs  $G_1$ ,  $G_2$  and  $G_3$  portrayed in Figure 6.11.

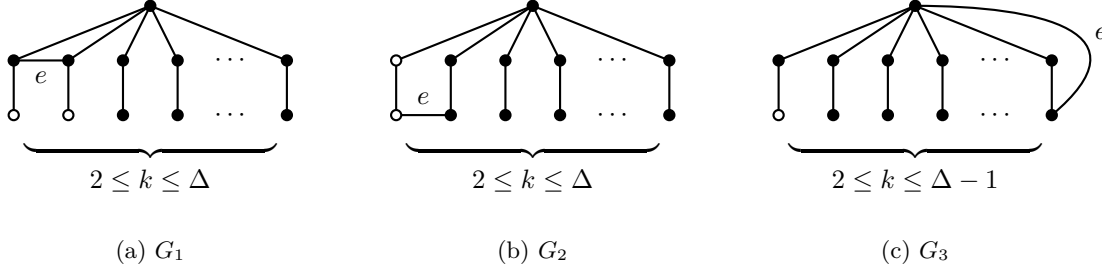


Figure 6.11: Possible graphs  $G \cong T_k + e$  as in Proposition 6.5

Suppose that  $G \cong G_1$ . In this case,  $n = 2k + 1$  and  $\gamma^{\text{OTD}}(G) \leq 2k - 1$ , where the shaded vertices in Figure 6.11(a) form an OTD-code of  $G$ . Thus,  $\gamma^{\text{OTD}}(G) \leq \left( \frac{2k-1}{2k+1} \right) n < \left( \frac{2\Delta-1}{2\Delta} \right) n$ .

Suppose that  $G \cong G_2$ . If  $k = 2$ , then  $G = C_5$ ,  $n = 5$  and  $\gamma^{\text{OTD}}(G) = 4 = \frac{4}{5}n < \left( \frac{2\Delta-1}{2\Delta} \right) n$ . If  $3 \leq k \leq \Delta$ , then  $n = 2k + 1$  and  $\gamma^{\text{OTD}}(G) \leq 2k - 1$ , where the shaded vertices in Figure 6.11(b) form an OTD-code of  $G$ . Thus,  $\gamma^{\text{OTD}}(G) \leq \left( \frac{2k-1}{2k+1} \right) n < \left( \frac{2\Delta-1}{2\Delta} \right) n$ .

Suppose that  $G \cong G_3$ . In this case,  $k \leq \Delta - 1$  noting that  $\Delta(G) \leq \Delta$ . As before,  $n = 2k + 1$  and  $\gamma^{\text{OTD}}(G) \leq 2k - 1$ , where the shaded vertices in Figure 6.11(c) form an OTD-code of  $G$ . Thus,  $\gamma^{\text{OTD}}(G) \leq \left( \frac{2k-1}{2k+1} \right) n < \left( \frac{2\Delta-1}{2\Delta} \right) n$ .  $\square$

We are now in a position to present a proof of our main result. Recall its statement.

**Theorem 6.5.** *For  $\Delta \geq 3$  a fixed integer, if  $G$  is an open-twin-free connected graph of order  $n \geq 5$  that contains no 4-cycles and satisfies  $\Delta(G) \leq \Delta$ , then*

$$\gamma^{\text{OTD}}(G) \leq \left( \frac{2\Delta - 1}{2\Delta} \right) n,$$

*except in one exceptional case when  $G = T_\Delta$ , in which case  $\gamma^{\text{OTD}}(G) = \left( \frac{2\Delta}{2\Delta+1} \right) n$ .*

*Proof.* We proceed by induction on the size  $m \geq 4$  of the connected graph  $G$ . If  $m = 4$ , then either  $G \cong P_5$  or  $G$  is isomorphic to a *paw*, that is a complete graph on three vertices (equivalently, a  $K_3$ ) with a leaf adjacent to one of the vertices of the  $K_3$ . Then,  $\gamma^{\text{OTD}}(G) = 4$  if  $G \cong P_5$  and  $\gamma^{\text{OTD}}(G) = 3$  if  $G$  is isomorphic to a *paw*. In both cases, we have  $\gamma^{\text{OTD}}(G) \leq \left( \frac{2\Delta}{2\Delta+1} \right) n$ . This establishes the base case. For the inductive hypothesis, let  $m \geq 5$  and assume that if  $G'$  is a connected graph  $G'$  of order  $n' \geq 5$  and size less than  $m$  that contains no 4-cycle and is open-twin-free and satisfies  $\Delta(G') \leq \Delta$ , then  $\gamma^{\text{OTD}}(G') \leq \left( \frac{2\Delta-1}{2\Delta} \right) n'$ , unless  $G' = T_\Delta$ .

We now consider the connected graph  $G$  of order  $n \geq 5$  and size  $m$  that contains no 4-cycle and is open-twin-free and satisfies  $\Delta(G) \leq \Delta$ . If the graph  $G$  is a tree, then the desired result follows from Theorem 6.6. Hence we may assume that the connected graph  $G$  contains a cycle. If  $m = 5$ , then the graph  $G$  is isomorphic to one of the graphs  $G_1$ ,  $G_2$  and  $G_3$  illustrated in Figure 6.11 and with  $k = 2$ , and the desired result holds by Proposition 6.5. Hence we may assume that  $m \geq 6$ .

Let  $C$  be an induced cycle in  $G$  and let  $e = uv$  be an edge of the cycle  $C$ . Let  $G' = G - e$  be the connected graph obtained from  $G$  by deleting the cycle edge  $e$ . The resulting graph  $G'$  has order  $n \geq 5$  and maximum degree  $\Delta(G') \leq \Delta(G) \leq \Delta$ . Since  $G$  contains no 4-cycle, neither does the graph  $G'$ . If  $G' \cong T_\Delta$ , then the graph  $G$  satisfies the statement of Proposition 6.5. Hence we may assume that  $G' \not\cong T_\Delta$ , for otherwise the desired result follows by Proposition 6.5.

Suppose that  $G'$  is open-twin-free. Applying the inductive hypothesis to the graph  $G'$ , there exists an OTD-code  $S'$  of  $G'$  such that  $|S'| \leq \left(\frac{2\Delta-1}{2\Delta}\right)n$ . We claim that  $S'$  is also an OTD-code of  $G$ . Suppose, to the contrary, that  $S'$  is not an OTD-code of  $G$ . In this case, at least one of  $u$  and  $v$  must be in the set  $S'$ . Renaming  $u$  and  $v$  if necessary, we may assume that  $u \in S'$  and that the vertex  $v$  is not identified by  $S'$  in  $G$ . Let  $x$  be the vertex in  $G$  such that  $v$  and  $x$  are identified by  $S'$  in  $G'$  but are not identified by  $S'$  in  $G$ . We note that the vertex  $x$  is neighbor of  $u$  different from  $v$ . Since  $S'$  is an OTD-code of  $G'$ , in order for  $S'$  to totally dominate the vertex  $v$  in  $G'$ , there exists a vertex  $w \in S'$  such that  $w \neq u$  and  $wv \in E(G)$ . Since the set  $S'$  does not identify the pair  $v$  and  $x$  in the graph  $G'$ , we infer that  $w \neq x$  and  $wx \in E(G)$ . This implies that  $G$  has a 4-cycle, namely,  $uvwxu$ , a contradiction. Therefore,  $S'$  is also an OTD-code of  $G$ , and so  $\gamma^{\text{OTD}}(G) \leq |S'| \leq \left(\frac{2\Delta-1}{2\Delta}\right)n$ .

Hence we may assume that  $G'$  has open twins, for otherwise the desired result follows. More generally, we may assume that for every cycle edge  $f$  of  $G$ , the graph  $G - f$  has open twins. Since  $G'$  has open twins, at least one of  $u$  and  $v$  is an open twin in  $G'$ . Renaming vertices if necessary, we may assume that  $v$  is an open twin in  $G'$ . Let  $x$  be an open twin of  $v$  in  $G'$ . Since  $G'$  has no 4-cycles, we infer that the open twins  $x$  and  $v$  must be of degree 1 in  $G'$ . Let  $w$  be the common neighbor of  $x$  and  $v$ . Thus,  $w$  is a support vertex in  $G'$  with  $v$  and  $x$  as leaf neighbors. Since  $e$  is a cycle edge of  $G$ , we note that the vertex  $w$  belongs to the cycle  $C$ . Thus,  $\deg_G(w) \geq 3$ ,  $\deg_G(x) = 1$ ,  $\deg_{G'}(v) = 1$  and  $\deg_G(v) = 2$ .

Let  $f = vw \in E(G)$  and let  $G'' = G - f$ . We note that the edge  $f$  belongs to the cycle  $C$ . By our earlier assumptions, the removal of the cycle edge  $f$  creates open twins. By our earlier observations, open twins in  $G''$  must be of degree 1 in  $G'$ . Since both  $u$  and  $w$  have degree at least 2 in  $G''$ , we therefore infer that the vertex  $v$  is an open twin in  $G''$ . Let  $y$  be an open twin of  $v$  in  $G''$ . Analogously as in the previous case when  $v$  is an open twin in the graph  $G'$ , we infer that the open twins  $v$  and  $y$  must be of degree 1 in  $G''$ . Thus, the vertex  $u$  is a support vertex in  $G''$  with  $v$  and  $y$  as leaf neighbors.

Continuing the same analysis for every edge of  $C$ , we infer that, if  $C: v_0v_1v_2 \dots v_pv_0$ , then renaming vertices of  $C$  if necessary, the vertex  $v_i$  is a support vertex in  $G$  for each odd  $i \in [p]$  and the vertex  $v_i$  has degree 2 in  $G$  for each even  $i \in [p]$ . Thus, we note  $p$  is an odd integer and, since  $G$  contains no 4-cycles,  $p$  must be at least 5. Therefore,  $C$  is a cycle of even length  $p+1 \geq 6$ . Let  $x_i$  be a leaf neighbor of the vertex  $v_i$  for each odd  $i \in [p]$  (see Figure 6.12). We note that for every odd  $i \in [p]$ , the vertex  $v_i$  has two neighbors on the cycle  $C$  (and these neighbors are distinct from the leaf  $x_i$ ), and so  $v_i$  has degree at least 3 in  $G$ . Moreover,  $m \geq p + p/2 = 3p/2 \geq 9$ .

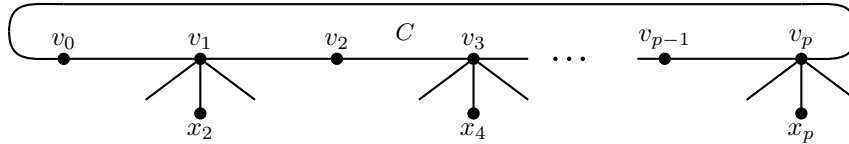


Figure 6.12: A possible structure of the graph  $G$  in the proof of Theorem 6.5

We now consider the graph  $G_0 = G - v_0$  obtained by deleting the vertex  $v_0$  from  $G$ . The graph  $G_0$  is a connected graph satisfying  $\Delta(G_0) \leq \Delta(G) \leq \Delta$ . Since  $G$  has no 4-cycles, the graph  $G_0$  also has no 4-cycles. Recall that the neighbors  $v_1$  and  $v_p$  of  $v_0$  are of degree at least 3 in  $G$  and are support vertices with leaf neighbors  $x_1$  and  $x_p$ , respectively. Since the graph  $G$  is open-twin-free, we therefore infer that the graph  $G_0$  has no open twins. Let  $G_0$  have size  $m_0$ , and so  $m_0 = m(G_0) = m - 2$ . By

our earlier observations,  $m \geq 9$ , and so  $m_0 \geq 7$ . Moreover since  $p \geq 5$ , the structure of the graph  $G$  implies that  $G_0 \not\cong T_\Delta$  (noting that  $x_1v_1v_2v_3x_3$  is a path in  $G_0$  where  $x_1$  and  $x_3$  are leaves in  $G_0$ , the vertex  $v_2$  has degree 2 in  $G_0$ , and  $v_3$  is a vertex of degree at least 3 in  $G_0$ ). Hence, by our induction hypothesis, there exists an OTD-code  $S_0$  of  $G_0$  such that  $|S_0| \leq \left(\frac{2\Delta-1}{2\Delta}\right)(n-1) < \left(\frac{2\Delta-1}{2\Delta}\right)n$ .

We show next that  $S_0$  is also an OTD-code of  $G$ . Since  $v_1$  and  $v_p$  are support vertices in  $G_0$ , we note that  $v_1, v_p \in S_0$ . To show that  $S_0$  is an OTD-code of  $G$ , we only need to show that  $S_0$  identifies the vertex  $v_0$  from every other vertex of  $G$ . However, since  $G$  has no 4-cycles, this is indeed true, as  $v_0$  is the only vertex of  $G$  with  $N_G(v_0) \cap S_0 = \{v_1, v_p\}$ . Hence, the result is true in this case as well. This completes the proof of Theorem 6.5.  $\square$

### 6.1.3.4 Tight examples

For  $\Delta \geq 3$  a fixed integer, let  $T_1$  and  $T_2$  be two vertex-disjoint copies of the reduced subdivided star  $T_\Delta^*$ . Let  $v_i$  be the central vertex in  $T_i$  (of degree  $\Delta$ ) and let  $u_i$  be the leaf neighbor of  $v_i$  for  $i \in [2]$ . Let  $T_{1,2}$  be the tree obtained from the union of  $T_1$  and  $T_2$  by adding the edge  $u_1u_2$ . The resulting tree  $T$  is illustrated in Figure 6.13 and satisfies  $\gamma^{\text{OTD}}(T) = \left(\frac{2\Delta-1}{2\Delta}\right)n$ . These examples show that the upper bound in Theorem 6.5 is best possible for every fixed value of  $\Delta \geq 3$ .

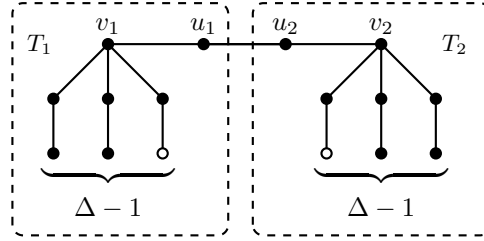


Figure 6.13: A tree  $T$  of order  $n$  satisfying  $\gamma^{\text{OTD}}(T) = \left(\frac{2\Delta-1}{2\Delta}\right)n$ .

A *subcubic graph* is a graph with maximum degree at most 3. In the special case when  $\Delta = 3$ , by Theorem 6.5 if  $G$  is a subcubic graph of order  $n \geq 5$  that is open-twin-free and contains no 4-cycles, then  $\gamma^{\text{OTD}}(G) \leq \frac{5}{6}n$ . We construct next a family of subcubic graphs  $G$  of arbitrarily large orders  $n$  that are open-twin-free and contain no 4-cycles satisfying  $\gamma^{\text{OTD}}(G) = \frac{5}{6}n$ . Let  $C: u_1u_2 \dots u_pu_1$  be a cycle of length  $p$  where  $p \geq 3$  and  $p \neq 4$ . For each vertex  $u_i$  on the cycle, we add a vertex disjoint copy,  $T_i$  say, of a reduced subdivided star  $T_3^*$  and identify one of the leaves at distance 2 from the central vertex of the reduced subdivided star with the vertex  $u_i$  for all  $i \in [p]$ . Let  $V(T_i) = \{u_i, v_i, w_i, x_i, y_i, z_i\}$  where  $u_iv_iw_ix_iy_i$  is a path and where  $z_i$  is the leaf neighbor of  $w_i$  in  $T_i$ . The resulting subcubic graph  $G_p$  is illustrated in Figure 6.14.

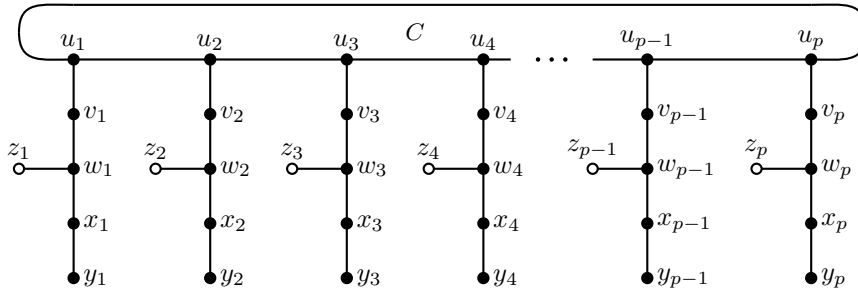


Figure 6.14: A subcubic graph  $G$  of order  $n$  satisfying  $\gamma^{\text{OTD}}(T) = \frac{5}{6}n$



**Proposition 6.6.** *For  $p \geq 3$  and  $p \neq 4$ , if the subcubic graph  $G_p$  has order  $n$ , then  $G_p$  is open-twin-free and contains no 4-cycles and satisfies  $\gamma^{\text{OTD}}(G_p) = \frac{5}{6}n$ .*

*Proof.* Let  $S$  be an arbitrary OTD-code in  $G_p$ . We show that  $|S \cap V(T_i)| \geq 5$  for all  $i \in [p]$ . Since  $S$  is a TD-set, the set  $S$  contains the support vertices  $w_i$  and  $x_i$ . In order to identify  $x_i$  and  $z_i$ , the vertex  $y_i$  belongs to the set  $S$ . In order to identify  $v_i$  and  $z_i$ , the vertex  $u_i$  belongs to the set  $S$ . In order to identify  $w_i$  and  $y_i$ , the set  $S$  contains at least one of  $v_i$  and  $z_i$ . Hence,  $|S \cap V(T_i)| \geq 5$  for all  $i \in [p]$ . Since  $S$  is an arbitrary OTD-code in  $G_p$  and  $|S| \geq \frac{5}{6}n$ , this implies that  $\gamma^{\text{OTD}}(T) \geq \frac{5}{6}n$ . Since the set  $S^* = (V(G_p) \setminus \{z_1, z_2, \dots, z_p\})$  is an OTD-code in  $G_p$ , we have  $\gamma^{\text{OTD}}(T) \leq \frac{5}{6}n$ . Consequently,  $\gamma^{\text{OTD}}(T) = \frac{5}{6}n$ .  $\square$

We remark that deleting the cycle edge  $u_1u_p$  in the construction of the subcubic graph  $G_p$  yields a tree  $T$  of order  $n$  that is open-twin-free and satisfies  $\gamma^{\text{OTD}}(T) = \frac{5}{6}n$ . By Proposition 6.6, the upper bound in Theorem 6.5 is tight for  $\Delta = 3$  in the strong sense that there exist connected subcubic graphs of arbitrary large order  $n$  that contain no 4-cycles, are open-twin-free, and that achieve the upper bound  $\gamma^{\text{OTD}}(G) = \left(\frac{2\Delta-1}{2\Delta}\right)n$ .

## 6.2 Open separation with domination

In this section, we study open-separating dominating codes of graphs. The said codes are rather new in the literature of identification problems and have been recently introduced by Chakraborty (the author) and Wagler in [52]. As we shall see, OD-codes are rather closely associated to OTD-codes by the fact that their respective code numbers on a graph can differ by at most one. In Chapter 8, we shall see that despite this closeness between the OD-number and the OTD-number, it is NP-hard to decide in general if the two numbers differ from or are equal to each other on a graph. In this section, we look at the problem of finding OD-codes of a graph as a covering problem in the OD-hypergraph (as discussed in Chapter 3) and discuss the polyhedra associated with OD-codes, again in relation to OTD-codes of some graph families already studied in this context.

We recall here that a graph  $G$  is OD-admissible if and only if  $G$  is open-twin-free. Most results discussed in this section also appear in [52].

### 6.2.1 Preliminary results

We begin with some general results on OD-codes and OD-numbers of graphs.

**Remark 6.3.** *Let  $G$  be an OD-admissible graph and let  $C$  be an OD-code of  $G$ . Then, there exists at most one vertex  $w$  of  $G$  such that  $N_G(w) \cap C = \emptyset$ .*

*Proof.* Toward contradiction, if there exist two distinct vertices  $u$  and  $v$  of  $G$  such that  $N_G(u) \cap C = N_G(v) \cap C = \emptyset$ , then  $C$  does not open-separate the pair  $(u, v)$ , a contradiction. This proves the result.  $\square$

**Remark 6.4.** *Let  $G$  be an OD-admissible graph and let  $C$  be a dominating set of  $G$  such that there exists at most one vertex  $w$  of  $G$  with  $N_G(w) \cap C = \emptyset$ . If the set  $C$  open-separates every pair  $(u, v)$  of distinct vertices of  $G$  with  $d_G(u, v) \leq 2$ , then  $C$  is an OD-code of  $G$ .*

*Proof.* It is enough to show that  $C$  open-separates every pair  $(u, v)$  of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . So, assume that  $u$  and  $v$  are two such vertices of  $G$  with  $d_G(u, v) \geq 3$ . Then, we have  $N_G(u) \Delta N_G(v) = N_G(u) \cup N_G(v)$ . If  $C$  does not open-separate the pair  $(u, v)$ , then it implies that  $C$  does not intersect  $N_G(u) \cup N_G(v)$ , that is,  $N_G(u) \cap C = N_G(v) \cap C = \emptyset$ , which contradicts our assumption. Therefore,  $C$  open-separates  $(u, v)$  and this proves the result.  $\square$

The next theorem brings out a close association between the OD- and the OTD-numbers of an OTD-admissible graph.

**Theorem 6.7** ([52]). *Let  $G$  be an OD-admissible graph. If  $G$  is a disjoint union of a graph  $G'$  and an isolated vertex, then we have*

$$\gamma^{\text{OD}}(G) = \gamma^{\text{OTD}}(G') + 1.$$

*Otherwise, we have*

$$\gamma^{\text{OTD}}(G) - 1 \leq \gamma^{\text{OD}}(G) \leq \gamma^{\text{OTD}}(G).$$

*Proof.* Let us first assume that  $G = G' \oplus \{v\}$  for some vertex  $v$  of  $G$ . Moreover, let  $C$  be any minimum OD-code of  $G$ . Then,  $v \in C$  in order for  $C$  to dominate  $v$ . This implies that  $N_G(v) \cap C = \emptyset$ . Therefore, by Remark 6.4, the set  $C \setminus \{v\}$  must total-dominate all vertices in  $V \setminus \{v\}$ , that is,  $C$  is an OD-code of  $G'$ . This implies that  $\gamma^{\text{OTD}}(G') \leq |C| - 1 = \gamma^{\text{OD}}(G) - 1$ , that is,  $\gamma^{\text{OD}}(G) \leq \gamma^{\text{OTD}}(G') + 1$ . On the other hand, if  $C'$  is an OTD-code of  $G'$ , then  $C' \cup \{v\}$  is an OD-code of  $G$ . This implies that  $\gamma^{\text{OD}}(G) \leq |C'| + 1 = \gamma^{\text{OTD}}(G') + 1$  and hence, the result follows.

Let us now assume that  $G$  has no isolated vertices. Then,  $G$  is also OTD-admissible. Now, the inequality  $\gamma^{\text{OD}}(G) \leq \gamma^{\text{OTD}}(G)$  holds by the fact that any OTD-code of  $G$  is also a dominating and an open-separating set of  $G$  and hence, is an OD-code of  $G$ . To prove the other inequality, we let  $C$  be a minimum OD-code of  $G$ . Then, we have  $|C| = \gamma^{\text{OD}}(G)$ . Now, if  $C$  is also an OTD-code of

$G$ , then the result holds trivially. Therefore, let us assume that  $C$  is not an OTD-code of  $G$ , that is, in particular,  $C$  is not a total-dominating set of  $G$ . This implies that there exists a vertex  $v$  of  $G$  such that  $C \cap N_G(v) = \emptyset$ . Moreover, by Remark 6.4,  $C \cap N_G(v) = \emptyset$  for exactly the vertex  $v$  of  $G$ . This implies that  $C \cup \{v\}$  is a total-dominating set of  $G$  and hence, is also an OTD-code of  $G$ . Therefore, we have  $\gamma^{\text{OTD}}(G) \leq |C| + 1 = \gamma^{\text{OD}}(G) + 1$ .  $\square$

The second relation between the OD- and the OTD-number of a graph in Theorem 6.7 was also presented in Corollary 3.6 of Section 3.3 in relation to the comparisons of code-number pairs depicted in Figure 3.2 of the same section.

**Remark 6.5.** *For a positive integer  $\ell \geq 2$ , there are infinitely many OD-admissible graphs  $G$  with components  $G_1, G_2, \dots, G_\ell$  such that*

$$\gamma^{\text{OD}}(G) - \sum_{i=1}^{\ell} \gamma^{\text{OD}}(G_i) = \ell - 1.$$

*Proof.* We prove the result by constructing the following example. Let  $G_1 \cong K_1$  and let  $G_i \cong K_2$  for all  $i \in [2, \ell]$ . Then it can be checked that  $\gamma^{\text{OD}}(G_i) = 1$  for all  $i \in [\ell]$ . Therefore, we have  $\sum_{i=1}^{\ell} \gamma^{\text{OD}}(G_i) = \ell$ . However, by Theorem 6.7, we have  $\gamma^{\text{OD}}(G) = 1 + \gamma^{\text{OTD}}(G - V(G_1)) = 1 + \sum_{i=2}^{\ell} \gamma^{\text{OTD}}(G_i)$ . Now, since  $\gamma^{\text{OTD}}(K_2) = 2$ , it implies that  $\gamma^{\text{OD}}(G) = 1 + 2(\ell - 1) = 2\ell - 1$ . This implies the result.  $\square$

Thus, Remark 6.5 shows that there exist non-connected graphs whose OD-numbers are strictly larger than the sum of the OD-numbers of its components. This in contrast to such results for X-codes with  $X \in \{\text{LD}, \text{LTD}, \text{ID}, \text{ITD}, \text{OTD}\}$  for which the X-number of the disjoint union of the components is equal to the sum of the X-numbers of the individual components.

## 6.2.2 OD-numbers of some graph families

In this section, we study the OD-numbers of graphs belonging to some well-known graph families. Moreover, motivated by Theorem 6.7 where the OD-number and the OTD-number differ by at most 1, we compare in the following the two numbers on some chosen graph families. This comparison also exhibits extremal cases for the upper bounds in Theorem 3.4.

### 6.2.2.1 Cliques and their disjoint unions

Cliques  $K_n$  are clearly open twin-free so that for  $n \geq 2$  both OD- and OLD-codes exist. It is known that the following holds.

$$\gamma^{\text{OTD}}(K_n) = \begin{cases} 2, & \text{if } n = 2; \\ n - 1, & \text{if } n \geq 3. \end{cases}$$

**Lemma 6.5** ([52]). *For a clique  $K_n$  with  $n \geq 2$ , we have  $\gamma^{\text{OD}}(K_n) = n - 1$ .*

*Proof.* Consider a clique  $K_n$  with  $n \geq 2$ . Then, the OD-hypergraph  $\mathcal{H}_{\text{OD}}(K_n)$  contains the following hyperedges.

- The closed neighborhoods  $N_{K_n}[v] = V(K_n)$  of all vertices  $v \in V(K_n)$ .
- The open-separators  $\triangle_O(K_n; u, v) = \{u, v\}$  of all distinct vertices  $u, v \in V(K_n)$ .

This shows that all hyperedges of  $\mathcal{H}_{\text{OD}}(K_n)$  of the type  $N_{K_n}[v]$  are redundant. Thus,  $\mathcal{C}_{\text{OD}}(K_n) = \mathcal{R}_n^2 = K_n$  follows which implies  $\gamma^{\text{OD}}(K_n) = \tau(\mathcal{R}_n^2) = n - 1$  by [7].  $\square$

Hence, the OD- and the OTD-numbers of cliques  $K_n$  differ only for  $n = 2$  and are equal for all  $n \geq 3$ . Consider now a graph  $G = K_{n_1} \oplus \dots \oplus K_{n_k}$  that is the disjoint union of  $k \geq 2$  cliques with  $1 < n_1 \leq \dots \leq n_k$ . It is well-known that the OTD-number of the disjoint union of two or more graphs is the sum of their OTD-numbers. Hence, we have

$$\gamma^{\text{OTD}}(K_{n_1} \oplus \dots \oplus K_{n_k}) = \sum_{n_i=2} 2 + \sum_{n_i \geq 3} (n_i - 1).$$

To compare this with the corresponding OD-numbers, we have the following.

**Lemma 6.6** ([52]). *Let  $G = K_{n_1} \oplus \dots \oplus K_{n_k}$  be a disjoint union of  $k \geq 2$  cliques with  $1 < n_1 \leq \dots \leq n_k$ .*

(a) *If  $n_1 = 2$ , then  $\gamma^{\text{OD}}(G) = -1 + \sum_{n_i=2} 2 + \sum_{n_i \geq 3} (n_i - 1)$ ,*

(b) *If  $n_1 \geq 3$ , then  $\gamma^{\text{OD}}(G) = \sum_{1 \leq i \leq k} (n_i - 1)$ .*

*Proof.* Consider the disjoint union  $G = K_{n_1} \oplus \dots \oplus K_{n_k}$  of cliques with  $1 < n_1 \leq \dots \leq n_k$  and  $k \geq 2$  and suppose that  $n_i = 2$  for all  $i \leq \ell \leq k$  (with possibly  $0 = \ell$ , i.e. all  $n_i \geq 3$ ). Let us further denote  $V_i = V(K_{n_i})$ . Then,  $\mathcal{H}_{\text{OD}}(G)$  is clearly composed of

- the closed neighborhoods  $N_G[v] = V_i$  of all vertices  $v \in V_i$ ,  $1 \leq i \leq k$  and
- the symmetric differences  $\Delta_O(G; u, v) = \{u, v\}$  of distinct vertices  $u, v \in V_i$  as well as  $\Delta_O(G; u, v) = N_G(u) \cup N_G(v)$  for  $v \in V_i$ ,  $u \in V_j$  with  $i \neq j$ .

This shows that all neighborhoods are redundant (by  $1 < n_1$ ), as well as all symmetric differences of open neighborhoods of vertices from different components as soon as at least one of them has order  $\geq 3$ . This implies  $\mathcal{C}_{\text{OD}}(G) = K_{2\ell} \oplus K_{n_{\ell+1}} \oplus \dots \oplus K_{n_k}$  and, accordingly,  $\gamma^{\text{OD}}(G) = (2\ell - 1) + \sum_{n_i \geq 3} (n_i - 1)$  follows by [7].  $\square$

Hence, for graphs  $G$  that are disjoint unions of cliques, the OD-numbers and the OTD-numbers are equal if all components are cliques of order  $\geq 3$ , but differ otherwise. In particular, if  $G$  is a *matching* (i.e. if  $n_i = 2$  for all  $1 \leq i \leq k$ ) and  $k \geq 2$ , the OTD-number of  $G$  is strictly greater than its OD-number, and the upper bound of  $\gamma^{\text{OD}}(G) = |V(G)| - 1$  from Theorem 3.4 is attained.

### 6.2.2.2 Bipartite graphs

Let  $B_k$  denote a half-graph on  $2k$  vertices. In [89] it was shown that the only graphs whose OTD-numbers equal the order of the graph are the disjoint unions of half-graphs. In particular, we have  $\gamma^{\text{OTD}}(B_k) = |V(B_k)| = 2k$ . Now, let  $G$  be a graph that is a disjoint union of half-graphs. Moreover, let  $G$  be on  $n$  vertices. Then, by Theorem 6.7, therefore, we have  $\gamma^{\text{OD}}(G) \geq \gamma^{\text{OTD}}(G) - 1 = n - 1$ . Moreover, by Theorem 3.4, we have  $\gamma^{\text{OD}}(G) \leq n - 1$ . Hence, combining the two inequalities, we get the following corollary.

**Corollary 6.2** ([52]). *For a graph  $G$  being the disjoint union of half-graphs, we have  $\gamma^{\text{OD}}(G) = |V(G)| - 1$ . In particular, for a half-graph  $B_k$ , we have  $\gamma^{\text{OD}}(B_k) = 2k - 1$ .*

Corollary 6.2 shows in particular that half-graphs and their disjoint unions are extremal examples of graphs whose OD-numbers also attain the general upper bound in Theorem 3.4.

**Characterizing graphs with OD-number equal to the order of graph.** Note that the upper bound from Theorem 3.4 does not apply to OD-admissible graphs having an isolated vertex. To see this, consider the graph  $G = B_k \oplus K_1$  for some  $k \geq 1$ . By Theorem 6.7, we have  $\gamma^{\text{OD}}(G) = \gamma^{\text{OTD}}(B_k) + 1 = 2k + 1 = |V(G)|$ . As a disjoint union of half-graphs, say

$$B_{k_1 k_2 \dots k_\ell} = B_{k_1} \oplus B_{k_2} \oplus \dots \oplus B_{k_\ell},$$



Figure 6.15: The black vertices depict an OD-code of the respective graph.

are the only graphs whose OTD-numbers equal the order of the graph by [89], the graphs of the form  $K_1 \oplus B_{k_1 k_2 \dots k_\ell}$  are the only ones whose OD-numbers equal the order of the graph.

Recall that a  $k$ -subdivided star is graph obtained from a  $k$ -star by subdividing the all edges of the star once. Therefore, a subdivided star  $k$ -subdivided star  $D_k$  is also a bipartite graph  $D_k = (U \cup W, E)$  with its stable vertex sets  $U = \{u_0, u_1, \dots, u_k\}$  and  $W = \{w_1, \dots, w_k\}$ , and edges  $u_i w_i$  and  $u_0 w_i$  for all  $w_i \in W$  (see Figure 6.15b). Then, we have  $D_1 = P_3$  and  $D_2 = P_5$ . Moreover, we clearly see that  $k$ -subdivided stars with  $k \geq 2$  are connected and open-twin-free and hence, are both OD- and OTD-admissible. As the next lemma shows,  $k$ -subdivided stars also provide examples of bipartite graphs where the OTD- and the OD-numbers disagree.

**Lemma 6.7** ([52]). *For a  $k$ -subdivided star  $D_k$  with  $k \geq 2$ , we have  $\gamma^{\text{OD}}(D_k) = 2k - 1$  and  $\gamma^{\text{OTD}}(D_k) = 2k$ .*

*Proof.* Let  $D_k = (U \cup W, E)$  with vertices  $U = \{u_0, u_1, \dots, u_k\}$ ,  $W = \{w_1, \dots, w_k\}$  and edges  $u_i w_i$  and  $u_0 w_i$  for all  $w_i \in W$ . Let  $C$  be an open-separating set of  $D_k$ . If there exist two vertices  $w_i, w_j$ ,  $1 \leq i < j$ , that do not belong to  $C$ , it implies that the vertices  $u_i$  and  $u_j$  are not open-separated, a contradiction. Therefore, we have  $|W \cap C| \geq k - 1$ . Similarly, if there exist two vertices  $u_i, u_j$ ,  $1 \leq i < j$ , that do not belong to  $C$ , it implies that the vertices  $w_i$  and  $w_j$  are not open-separated, again a contradiction. This implies that  $|U \cap C| \geq k - 1$ . Hence, since  $U$  and  $W$  are disjoint sets, we have  $|C| \geq 2k - 2$ . Now, if  $|C| = 2k - 2$  exactly, it would imply that  $C = U \cup W \setminus \{u_0, u_i, w_j\}$  for some  $1 \leq i, j \leq k$ . However, this would mean that  $N_{D_k}(w_i) \cap C = N_{D_k}(u_j) \cap C = \emptyset$  and thus, the pair  $(u_j, w_i)$  are not open-separated by  $C$ , a contradiction. This implies that  $|C| \geq 2k - 1$ . Thus, we have  $\gamma^{\text{OD}}(D_k) \geq 2k - 1$ . Moreover, it can be verified that the set  $U \cup W \setminus \{u_0, w_1\}$  is an OD-code of  $D_k$  of order  $2k - 1$ . Hence, we have  $\gamma^{\text{OD}}(D_k) = 2k - 1$ .

By Theorem 6.7, we have  $\gamma^{\text{OTD}}(D_k) \geq 2k - 1$ . However, if  $\gamma^{\text{OTD}}(D_k) = 2k - 1$  exactly and  $C$  is a minimum OTD-code of  $D_k$ , then we have  $C = U \cup W \setminus \{w_i\}$  for some  $1 \leq i \leq k$  (since we must have  $U \subset C$ ). However, this would mean that the vertex  $u_i$  is not open-dominated by  $C$ . Hence, we must have  $\gamma^{\text{OTD}}(D_k) \geq 2k$ . Moreover, it can be verified that the set  $U \cup W \setminus \{u_0\}$  is an OTD-code of  $D_k$  of order  $2k$ . This proves the result.  $\square$

### 6.2.2.3 Split graphs

In order to study OD-codes of split graphs and compare them with the OTD-codes, we restrict ourselves to split graphs  $G$  without open twins and isolated vertices. To that end, we look at the thin and the thick headless spiders. It is easy to check that the thin and the thick headless spiders have no twins. In [11], it was shown that  $\gamma^{\text{OTD}}(H_k) = k$  for  $k \geq 3$  and  $\gamma^{\text{OTD}}(\overline{H}_k) = k + 1$  for  $k \geq 3$ . We next analyze the OD-numbers of the thin and the thick headless spiders.

**Lemma 6.8** ([52]). *For any integer  $k \geq 3$  and thin and thick headless spiders  $H_k$  and  $\overline{H}_k$ , respectively, we have  $\gamma^{\text{OD}}(H_k) = k$  and  $\gamma^{\text{OD}}(\overline{H}_k) = k + 1$ .*

*Proof.* Let  $H_k = (Q \cup S, E)$ , where  $Q = \{q_1, \dots, q_k\}$  is a clique and  $S = \{s_1, \dots, s_k\}$  is an independent set. Let us first show that  $\gamma^{\text{OD}}(H_k) = k$ . Let  $C$  be a minimum OD-code of  $H_k$ . If  $C$  is also a total-dominating set, then  $C$  is an OTD-code and hence, we are done by the fact that  $\gamma^{\text{OTD}}(H_k) = k$  proved in [11]. So, let us assume that  $C$  is not a total-dominating set of  $H_k$ . This, along

with Remark 6.4, implies that there exists exactly one vertex  $v$  of  $H_k$  such that  $C \cap N_{H_k}(v) = \emptyset$ . Let us first assume that  $v \in Q$  and that  $v = q_1$ , without loss of generality. This implies that  $\{q_2, q_3\} \cap C = \emptyset$  (notice that the vertices  $q_2$  and  $q_3$  exist since  $k \geq 3$ ). This further implies that  $N_{H_k}(s_2) \cap C = N_{H_k}(s_3) \cap C = \emptyset$ , a contradiction. Let us therefore assume that  $v \in S$ . Again, let  $v = s_1$ , without loss of generality. Then, by the uniqueness of the vertex  $s_1$  with respect to the constraint  $C \cap N_{H_k}(s_1) = \emptyset$ , we must have  $q_2, q_3, \dots, q_k \in C$ . Also, recall that  $v = s_1 \in C$ . This implies that  $|C| \geq k$ . Since  $\gamma^{\text{OTD}}(H_k) = k$  by results in [11], we have  $\gamma^{\text{OD}}(H_k) = \gamma^{\text{OTD}}(H_k) = k$ .

We now show that  $\gamma^{\text{OD}}(\overline{H}_k) = k + 1$ . If on the contrary,  $\gamma^{\text{OTD}}(\overline{H}_k) = \gamma^{\text{OD}}(\overline{H}_k) + 1$ , then any minimum OD-code  $C$  of  $\overline{H}_k$  is not a total-dominating set of  $G$  and hence, by Remark 6.4, there exists exactly one vertex  $v$  of  $\overline{H}_k$  such that  $C \cap N_{H_k}(v) = \emptyset$ . Let us first assume that  $v \in Q$  and that  $v = q_1$ , without loss of generality. This implies that  $(S \setminus \{s_1\}) \cap C = \emptyset$ . This further implies that  $C$  does not intersect  $N_{H_k}(q_1) \triangle N_{H_k}(s_1) = S \setminus \{s_1\}$ , that is,  $C$  does not open-separate the pair  $(q_1, s_1)$ , a contradiction. Let us therefore assume that  $v \in S$ . Again, let  $v = s_1$ , without loss of generality. Then,  $\{q_2, q_3\} \cap C = \emptyset$  (notice that the vertices  $q_2$  and  $q_3$  exist since  $k \geq 3$ ). This implies that  $C$  does not intersect  $N_{H_k}(s_2) \triangle N_{H_k}(s_3) = \{u_2, u_3\}$ , that is  $C$  does not open-separate the pair  $(s_2, s_3)$ , again a contradiction. Hence, this proves that  $\gamma^{\text{OTD}}(\overline{H}_k) = \gamma^{\text{OD}}(\overline{H}_k) = k + 1$  (the second equality is by using the result for  $\gamma^{\text{OTD}}(\overline{H}_k)$  in [11]).  $\square$

Hence, Lemma 6.8 combined with the results from [11] show that for the thin and the thick headless spiders  $H_k$  and  $\overline{H}_k$ , respectively, the OD- and the OTD-numbers are equal for all  $k \geq 3$ . It would be interesting to study whether there exist families of open-twin-free split graphs where the OD- and the OTD-numbers differ.

### 6.2.2.4 Thin suns

The result in Lemma 6.8 on thin headless spiders can be further generalized to thin suns. Let  $T_k = (C \cup S, E)$  be a thin sun, where  $C$  is a clique and  $S$  is an independent set of  $T_k$ . Then, we call two vertices  $c_i, c_j \in C$  of the thin sun  $T_k$  *open  $C$ -twins* if  $c_i$  and  $c_j$  are non-adjacent and  $N_{T_k}(c_i) \cap C = N_{T_k}(c_j) \cap C$ . For instance, the sunlet in Figure 2.11(a) and the thin sun in Figure 2.11(b) have open  $C$ -twins, whereas the thin headless spider in Figure 2.11(c) does not.

In [11], it was shown that for a thin sun  $T_k$  with  $k \geq 4$  and without open  $C$ -twins, the set  $C$  is the unique minimum OTD-code of  $T_k$  and thus, we have  $\gamma^{\text{OTD}}(T_k) = k$ . Now, with regards to OD-numbers of thin suns, we can show that  $\gamma^{\text{OD}}(T_k) = k$ . However, before that, we prove the following lemma.

**Lemma 6.9.** *For a thin sun  $T_k = (C \cup S, E)$  with  $k \geq 4$ , the OD-clutter  $\mathcal{C}_{\text{OD}}(T_k)$  is composed of*

- $N_{T_k}[s_i] = \{s_i, c_i\}$  for all  $s_i \in S$ ,
- $\Delta_O(T_k; s_i, s_j) = \{c_i, c_j\}$  for all distinct  $s_i, s_j \in S$ ,
- $\Delta_O(T_k; c_i, c_j) = \begin{cases} \{s_i, s_j\}, & \text{if } c_i, c_j \text{ are open } C\text{-twins;} \\ \{s_i, s_j, c_\ell\}, & \text{if } c_i, c_j \text{ non-adjacent and } \{c_\ell\} = \Delta_O(T_k, c_i, c_j) \cap C, \end{cases}$
- $\Delta_O(T_k; c_i, s_j) = \{s_i, c_\ell\}$  if  $c_i, c_j$  are adjacent and  $\{c_\ell\} = N_{T_k}(c_i) \cap C \setminus \{c_j\}$ ,  $\ell \neq i, j$ .

*Proof.*  $\mathcal{H}_{\text{OD}}(T_k)$  is composed of the closed neighborhoods

- $N_{T_k}[s_i] = \{s_i, c_i\}$  for all  $s_i \in S$ ,
- $N_{T_k}[c_i] = N_{T_k}[c_i] \cap C \cup \{s_i\}$  for all  $c_i \in C$

and the symmetric differences

- $\Delta_O(T_k; s_i, s_j) = \{c_i, c_j\}$  for all distinct  $s_i, s_j \in S$ ,
- $\Delta_O(T_k; c_i, c_j) = \Delta_O(T_k; c_i, c_j) \cap C \cup \{s_i, s_j\}$  for all distinct  $c_i, c_j \in C$ ,

- $\Delta_O(T_k; c_i, s_j) = \begin{cases} N_{T_k}[c_i], & \text{if } i = j; \\ (N_C(c_i) \cup \{s_i\}) \triangle \{c_j\}, & \text{if } i \neq j. \end{cases}$

This shows that  $N_{T_k}[s_i]$  and  $\Delta_O(T_k; s_i, s_j)$  belong to  $\mathcal{C}_{OD}(T_k)$  which further implies that the following hyperedges from  $\mathcal{H}_{OD}(T_k)$  are redundant:

- $N_{T_k}[c_i] = N_C[c_i] \cup \{s_i\}$  for all  $c_i \in C$ ,
- $\Delta_O(T_k; c_i, c_j)$  if  $|\Delta_O(T_k; c_i, c_j) \cap C| \geq 2$ ,
- $\Delta_O(T_k; c_i, s_j)$  if  $c_i, c_j$  are non-adjacent or if  $c_i, c_j$  are adjacent but  $|N_C(c_i)| \geq 3$ .

In the remaining cases,  $\Delta_O(T_k; c_i, c_j)$  and  $\Delta_O(T_k; c_i, s_j)$  belong to  $\mathcal{C}_{OD}(T_k)$ .  $\square$

**Lemma 6.10** ([52]). *For a thin sun  $T_k = (C \cup S, E)$  with  $k \geq 4$  and without open  $C$ -twins, the set  $C$  is a minimum OD-code and hence, we have  $\gamma^{OD}(T_k) = |C| = k$ .*

*Proof.* By Lemma 6.9, the OD-clutter  $\mathcal{C}_{OD}(T_k)$  contains  $\{s_i, c_i\}$  for all  $s_i \in S$ , which implies the lower bound  $\gamma^{OD}(T_k) \geq k$ . On the other hand, it is easy to verify that, by Lemma 6.9, all hyperedges of the OD-clutter  $\mathcal{C}_{OD}(T_k)$  have a non-empty intersection with  $C$  (when  $T_k$  has no open  $C$ -twins) so that  $C$  is a cover of  $\mathcal{C}_{OD}(T_k)$  of order  $k$ . Hence,  $C$  is a minimum OD-code and the assertion  $\gamma^{OD}(T_k) = |C| = k$  follows.  $\square$

Therefore, thin suns without open  $C$ -twins are examples of graphs where the OD- and the OTD-number are equal. This applies in particular to sunlets  $T_k$  with  $k \geq 5$  and to thin headless spiders. However, for thin suns  $T_k$  with open  $C$ -twins,  $\gamma^{OD}(T_k)$  and  $\gamma^{OTD}(T_k)$  may differ. For instance, for the thin sun  $T_4$  depicted in Fig. 2.11(b), it can be checked that  $\gamma^{OD}(T_4) = 4 < 5 = \gamma^{OTD}(T_4)$ . We call a thin sun  $T_k = (C \cup S, E)$  *almost complete* if  $k = 2\ell$  and  $c_i$  is non-adjacent to  $c_{i+\ell}$  but is adjacent to all other  $c_j \in C$ . We can show the following.

**Lemma 6.11** ([52]). *For an almost complete thin sun  $T_{2\ell}$  with  $\ell \geq 3$ , we have  $\gamma^{OD}(T_{2\ell}) = 3\ell - 1$  and  $\gamma^{OTD}(T_{2\ell}) = 3\ell$ .*

*Proof.* In  $T_{2\ell}$ , the vertices  $c_i, c_{i+\ell}$  form open  $C$ -twins. By Lemma 6.9, the OD-clutter  $\mathcal{C}_{OD}(T_{2\ell})$  is composed of

- $N_{T_{2\ell}}[s_i] = \{s_i, c_i\}$  for all  $s_i \in S$ ,
- $\Delta_O(T_{2\ell}; s_i, s_j) = \{c_i, c_j\}$  for all distinct  $s_i, s_j \in S$ ,
- $\Delta_O(T_{2\ell}; c_i, c_{i+\ell}) = \{s_i, s_{i+\ell}\}$  for all  $c_i \in C$ .

Hence, every cover of  $\mathcal{C}_{OD}(T_{2\ell})$  has to contain at least all but one vertex from  $C$  and one of  $\{s_i, s_{i+\ell}\}$  for all  $1 \leq i \leq \ell$ , showing that  $\gamma^{OD}(T_{2\ell}) \geq 3\ell - 1$ . We next observe that  $V' = \{s_1, \dots, s_\ell\} \cup C \setminus \{c_j\}$  for some  $j \in \{1, \dots, \ell\}$  is a cover of  $\mathcal{C}_{OD}(T_{2\ell})$ . Indeed,  $V'$  meets all

- $N[s_i] = \{s_i, c_i\}$  in  $s_i$  for  $1 \leq i \leq \ell$  and in  $c_i$  for  $\ell + 1 \leq i \leq 2\ell$ ,
- $\Delta_O(T_{2\ell}; s_i, s_j) = \{c_i, c_j\}$  as  $V'$  contains all but one vertex from  $C$ ,
- $\Delta_O(T_{2\ell}; c_i, c_{i+\ell}) = \{s_i, s_{i+\ell}\}$  in  $s_i$  for  $1 \leq i \leq \ell$ .

Hence,  $V'$  is an OD-code of  $T_{2\ell}$  and  $\gamma^{OD}(T_{2\ell}) = 3\ell - 1$  follows. Furthermore, from [11] we deduce that the OTD-clutter  $\mathcal{C}_{OTD}(T_{2\ell})$  is composed of

- $N(s_i) = \{c_i\}$  for all  $s_i \in S$ ,
- $\Delta_O(T_{2\ell}; c_i, c_{i+\ell}) = \{s_i, s_{i+\ell}\}$  for all  $c_i \in C$ .

For an almost complete thin sun  $T_{2\ell}$  with  $\ell \geq 3$ , this implies that  $C \cup \{s_1, \dots, s_\ell\}$  is a minimum OTD-code of  $T_{2\ell}$  and, hence,  $\gamma^{OTD}(T_{2\ell}) = 3\ell$  follows.  $\square$

Hence, there exist infinitely many thin suns with open  $C$ -twins for which the OD- and the OTD-numbers differ. Determining the OD-clutters of the graph families studied below showed their relation to different complete  $q$ -roses. Relying on the results from [7] on polyhedra associated to complete  $q$ -roses enabled us to prove the following.

### 6.2.2.5 OD-polyhedra

In this subsection, we investigate the OD-clutters and the OD-polyhedra of graphs. In particular, we look at cliques and headless spiders.

**Theorem 6.8.** *Let  $G = (V, E)$  be either a clique  $K_n$  with  $n \geq 2$  or a matching  $kK_2$  with  $k \geq 1$  and  $n = 2k$ . Then, we have  $\mathcal{C}_{\text{OD}}(G) = \mathcal{R}_n^2 = K_n$  and  $P_{\text{OD}}(G)$  is given by*

- (a) *a nonnegativity constraint  $x_v \geq 0$  for all vertices  $v \in V$  and*
- (b)  *$x(V') = \sum_{v \in V'} x_v \geq |V'| - 1$  for all subsets  $V' \subseteq V$  with  $|V'| \geq 2$ .*

*Proof.* From the proofs of Lemma 6.5 and Lemma 6.6, respectively, we see that  $\mathcal{C}_{\text{OD}}(K_n) = \mathcal{R}_n^2 = K_n$  and  $\mathcal{C}_{\text{OD}}(kK_2) = \mathcal{R}_{2k}^2 = K_{2k}$  holds. Hence,  $\mathcal{C}_{\text{OD}}(G)$  is in both cases a complete 2-rose of order  $n$  and  $P_{\text{OD}}(G)$  is accordingly given by nonnegativity constraints for all vertices  $v \in V$  and constraints  $x(V') \geq |V'| - 1$  for all subsets  $V' \subseteq V$  with  $|V'| \geq 2$  by Equation (3.1).  $\square$

Note that two graphs with equal OD-clutters have the same set of OD-codes and thus also the same OD-numbers and the same OD-polyhedra. Theorem 6.8 shows that this applies to cliques and matchings.

**Theorem 6.9** ([52]). *Let  $\overline{H}_k = (Q \cup S, E)$  be a thick headless spider with  $k \geq 4$ , where  $Q$  is a clique and  $S$  is an independent set with  $|Q| = |S| = k$ . Then, we have  $\mathcal{C}_{\text{OD}}(\overline{H}_k) = \mathcal{R}_{|S|}^{|S|-1} \cup \mathcal{R}_{|Q|}^2$  and  $P_{\text{OD}}(\overline{H}_k)$  is given by the constraints*

- (a)  $x_v \geq 0$  for all vertices  $v \in Q \cup S$ ,
- (b)  $x(V') = \sum_{v \in V'} x_v \geq |V'| - k + 2$  for all  $V' \subseteq S$  with  $|V'| \geq k - 1$ ,
- (c)  $x(V') = \sum_{v \in V'} x_v \geq |V'| - 1$  for all  $V' \subseteq Q$  with  $|V'| \geq 2$ .

*Proof.* Let  $Q = \{q_1, q_2, \dots, q_k\}$  and  $S = \{s_1, s_2, \dots, s_k\}$ . Then, the hypergraph  $\mathcal{H}_{\text{OD}}(\overline{H}_k)$  contains the following neighborhoods of vertices of  $\overline{H}_k$  as its hyperedges.

- $N_{\overline{H}_k}[s_i] = Q \setminus \{q_i\} \cup \{s_i\}$  for all  $s_i \in S$ ,
- $N_{\overline{H}_k}[q_i] = Q \cup \{s_i\}$  for all  $q_i \in Q$ .

$\mathcal{H}_{\text{OD}}(\overline{H}_k)$  also contains the following separators as its hyperedges.

- $\Delta_{\text{O}}(\overline{H}_k; s_i, s_j) = \{q_i, q_j\}$  for all distinct  $s_i, s_j \in S$ ,
- $\Delta_{\text{O}}(\overline{H}_k; q_i, q_j) = \{q_i, q_j\} \cup \{s_i, s_j\}$  for all distinct  $q_i, q_j \in Q$ ,
- $\Delta_{\text{O}}(\overline{H}_k; q_i, s_j) = \begin{cases} S \setminus \{s_i\}, & \text{if } i = j; \\ \{q_i, q_j\} \cup S \setminus \{s_i\}, & \text{if } i \neq j. \end{cases}$

This shows that all neighborhoods are redundant as well as all  $\Delta_{\text{O}}(\overline{H}_k; q_i, q_j)$  and  $\Delta_{\text{O}}(\overline{H}_k; q_i, s_j)$  with  $i \neq j$  so that only  $\Delta_{\text{O}}(\overline{H}_k; s_i, s_j)$  and  $\Delta_{\text{O}}(\overline{H}_k; q_i, s_i)$  belong to  $\mathcal{C}_{\text{OD}}(\overline{H}_k)$ . Hence, we obtained  $\mathcal{C}_{\text{OD}}(\overline{H}_k) = \mathcal{R}_{|S|}^{|S|-1} \cup \mathcal{R}_{|Q|}^2$ . Again, applying the result on polyhedra associated to complete  $q$ -roses from [7] shows that  $P_{\text{OD}}(\overline{H}_k)$  is given by nonnegativity constraints for all vertices and the constraints

- $x(V') = \sum_{v \in V'} x_v \geq |V'| - k + 2$  for all  $V' \subseteq S$  with  $|V'| \geq k - 1$ ,
- $x(V') = \sum_{v \in V'} x_v \geq |V'| - 1$  for all  $V' \subseteq Q$  with  $|V'| \geq 2$ .

$\square$

Comparing this result with the result from [11] on OTD-codes of thick headless spiders, we observe that  $\mathcal{C}_{\text{OD}}(\overline{H}_k) = \mathcal{C}_{\text{OTD}}(\overline{H}_k)$ . Hence, a vertex subset is an OD-code of  $\overline{H}_k$  if and only if it is an OTD-code of  $\overline{H}_k$ . Accordingly, the OD- and the OTD-numbers and as well as the OD- and the OTD-polyhedra are equal for thick headless spiders.



**Theorem 6.10** ([52]). *Let  $H_k = (Q \cup S, E)$  be a thin headless spider with  $k \geq 4$ , where  $Q$  is a clique and  $S$  is an independent set with  $|Q| = |S| = k$ . Then, we have  $\mathcal{C}_{\text{OD}}(H_k) = H_k$  and  $P_{\text{OD}}(H_k)$  is given by the constraints*

- (a)  $x_v \geq 0$  for all vertices  $v \in Q \cup S$ ,
- (b)  $x_{q_i} + x_{s_i} \geq 1$  for  $1 \leq i \leq k$ ,
- (c)  $x(V') = \sum_{v \in V'} x_v \geq |V'| - 1$  for all  $V' \subseteq Q$  with  $|V'| \geq 2$ .

*Proof.* Let  $Q = \{q_1, q_2, \dots, q_k\}$  and  $S = \{s_1, s_2, \dots, s_k\}$ . Recall that  $H_k$  is a special thin sun  $T_k = (Q \cup S, E)$ . From Lemma 6.9, we deduce that in  $\mathcal{C}_{\text{OD}}(H_k)$  we only have

- $N_{H_k}[s_i] = \{s_i, q_i\}$  for all  $s_i \in S$ ; and
- $\Delta_O(H_k; s_i, s_j) = \{q_i, q_j\}$  for all distinct  $s_i, s_j \in S$ .

Thus, we have  $\mathcal{C}_{\text{OD}}(H_k) = H_k$ . Hence  $\mathcal{C}_{\text{OD}}(H_k)$  is composed of the matching  $\{q_i, s_i\}$  for  $1 \leq i \leq k$  and the complete 2-rose  $\mathcal{R}_{|Q|}^2$ . By  $\mathcal{F}_{\text{OD}}^1(H_k) = \emptyset$ , we have nonnegativity constraints for all vertices. Moreover,  $P_{\text{OD}}(H_k)$  has clearly the constraints  $x(V') \geq |V'| - 1$  for all  $V' \subseteq Q$  with  $|V'| \geq 2$  by [7]. Finally,  $x_{q_i} + x_{s_i} \geq 1$  define facets for  $1 \leq i \leq k$  by [16], and it is easy to see that no further constraints are needed to describe  $P_{\text{OD}}(H_k)$  (as each  $x_{s_i}$  occurs in exactly one constraint different from a nonnegativity constraint).  $\square$

Combining all the constraints in (b) yields  $x(Q) + x(S) \geq k$  and this implies  $\gamma^{\text{OD}}(H_k) \geq k$ . It is also easy to see that  $Q$  is a cover of  $\mathcal{C}_{\text{OD}}(H_k)$  and hence,  $\gamma^{\text{OD}}(H_k) = k$ . This illustrates how, on the one hand, polyhedral arguments can be used to determine lower bounds for OD-numbers and, on the other hand, an analysis of the OD-clutter provides OD-codes. Moreover, if the order of the latter meets the lower bound, the OD-number of the studied graph is determined.

We note further that manifold hypergraphs have been already studied in the covering context, see, for example, [4, 72, 169] to mention just a few. The same techniques as illustrated above with the help of complete  $q$ -roses can be applied whenever the OD-clutter of some graph equals such a hypergraph or contains such a hypergraph as substructure, which gives an interesting perspective of studying OD-polyhedra (and other X-polyhedra as well, for  $X \neq \text{OD}$ ) further.

## 6.3 Conclusion

In this chapter, we study open separation in graphs on some graph families and also address for these graph families some conjectures from the literature. In what follows, we summarize our work with respect to the different sections in this chapter and also raise some relevant questions as a future line of research.

### 6.3.1 OTD-codes of block graphs

OTD-CODE can be solved in linear time when the input graph is a block graph [10]. In our study, we complement this result by presenting tight lower and upper bounds for the OTD-codes of block graphs. We gave bounds both in terms of the number of vertices and the number of blocks of a block graph  $G$ . In Theorem 6.1, we also generalized to block graphs an existing result by Foucaud et al. in [89] characterizing the set of all extremal graphs to be half-graphs.

The structural properties of block graphs have enabled us to prove interesting bounds for their OTD-numbers. It would be interesting to see whether other structured families can be studied in a similar way. Noting that block graphs are a subfamily of chordal graphs, it is natural to ask the following question.

**Open Problem 6.1.** *What are the tight upper and lower bounds on the OTD-numbers of OTD-admissible chordal graphs?*

### 6.3.2 OTD-codes of $C_4$ -free graphs of given maximum degree

Our main result, namely Theorem 6.5, shows that for  $\Delta \geq 3$  a fixed integer, if  $G$  is a connected graph of order  $n \geq 5$  that is open-twin-free and satisfies  $\Delta(G) \leq \Delta$ , then  $\gamma^{\text{OTD}}(G) \leq \left(\frac{2\Delta-1}{2\Delta}\right)n$ , except in one exceptional case when  $G$  is a subdivided star of maximum degree  $\Delta$ , that is, if  $G = T_\Delta$ . As shown in Proposition 6.3, if  $G = T_\Delta^*$  is the reduced subdivided star of order  $n$ , then  $\gamma^{\text{OTD}}(G) = \left(\frac{2\Delta-1}{2\Delta}\right)n$ .

As remarked earlier, the upper bound in Theorem 6.5 is best possible when  $\Delta = 3$ , for arbitrarily large connected graphs. For every fixed value of  $\Delta \geq 4$ , it remains an open problem to determine if the upper bound in Theorem 6.5 is tight for arbitrarily large connected graphs, or if it can be improved for connected graphs of sufficiently large order  $n$ .

When we consider general graphs and thus allow 4-cycles, the bound of Theorem 6.5 does not hold. In fact, as shown in [89], the infinite family of half-graphs provides infinitely many connected bipartite graphs  $G$  of order  $n$  with  $\gamma^{\text{OTD}}(G) = n$ . By using these graphs as building blocks, one can build, for every fixed value of  $\Delta$ , arbitrarily large connected graphs  $G$  of order  $n$  with  $\gamma^{\text{OTD}}(G) = \left(\frac{2\Delta}{2\Delta+1}\right)n$  (similar to the graphs constructed in Proposition 6.6). Thus, one may ask the following question.

**Open Problem 6.2.** *What is the best possible bound of this form in the general case (when one excludes half-graphs)?*

We shall investigate this problem in future work. We also recall here two questions by Henning and Yeo from [130] about regular graphs. First, they conjectured the following.

**Open Problem 6.3** (Conjecture [130]). *For every connected open-twin-free cubic graph  $G$ , the upper bound  $\gamma^{\text{OTD}}(G) \leq \frac{3}{5}n$  holds, except for a finite number of graphs.*

Second, they also asked the following question.

**Open Problem 6.4.** *Is it true that  $\gamma^{\text{OTD}}(G) \leq \left(\frac{\Delta}{\Delta+1}\right)n$  holds for every connected open-twin-free  $\Delta$ -regular graph  $G$  of order  $n$ ?*

### 6.3.3 OD-codes of graphs

In our work on OD-codes in this chapter, we showed, as a general result, that the OD- and the OTD-number of a graph differ by at most one. This motivated us to compare the two numbers on several graph families. This study revealed that they

- are equal, for example, for cliques  $K_n$  with  $n \geq 3$ , thin and thick headless spiders  $H_k$  and  $\overline{H}_k$ , respectively, with  $k \geq 3$ , and thin suns  $T_k = (C \cup S, E)$  with  $k \geq 4$  and without open  $C$ -twins;
- differ for example, for matchings  $kK_2$  with  $k \geq 1$ , half-graphs  $B_k$  with  $k \geq 1$  and their disjoint unions,  $k$ -subdivided stars  $D_k$  with  $n \geq 2$ , and almost complete thin suns  $T_{2\ell}$  with  $\ell \geq 3$ .

In particular, this showed that the OD-numbers of cliques, half-graphs and their disjoint unions attain the upper bound in Theorem 3.4.

A future line of research on OD-codes would include studying them on more graph families. It would also be interesting to investigate further to see for which graph families, the OD- and the OTD-numbers of its members differ and for which they are equal.



# Chapter 7

## Full separation in graphs

The concept of full separation is rather new in the literature of identification problems and was recently introduced by Chakraborty (the author) and Wagler in [53]. In this section, we study this separation property in combination with dominating sets (that is, FD-codes) and total-dominating set (that is, FTD-codes). Our principal objective is to study the FD- and FTD-numbers of graphs belonging to some well-known graph families. In one of the general results about the two codes, we see (in Theorem 7.1) that the FD- and the FTD-numbers of a graph can differ by at most 1. This motivates us to compare the two numbers for different families of graphs.

To start with, we recall that a graph  $G$  is full-separable if and only if  $G$  is twin-free. Hence, all graphs in this chapter on which either the FD- or the FTD-codes are studied are assumed to be twin-free. Now, using the definition related to full separation and Observations 2.5 and 2.6, the next remark captures (all at one place) the equivalent conditions to imply when a vertex subset of a full-separable graph  $G$  is a full-separating set of  $G$ .

**Remark 7.1.** *Let  $G$  be a twin-free graph and let  $C$  be a vertex subset of  $G$ . Then, the following assertions are equivalent.*

- (1)  $C$  is a full-separating set.
- (2)  $C$  is both a closed-separating and an open-separating set of  $G$ .
- (3)  $N_G[u] \cap C \neq N_G[v] \cap C$  and  $N_G(u) \cap C \neq N_G(v) \cap C$  for each pair of distinct vertices  $u, v \in V(G)$ .
- (4)  $C$  has a non-empty intersection with  $\Delta_F(G; u, v) = \Delta_O(G; u, v) \setminus \{u, v\} = \Delta_C(G; u, v) \setminus \{u, v\}$  for all distinct  $u, v \in V(G)$ .
- (5)  $C$  has a non-empty intersection with
  - (a)  $N_G(u) \triangle N_G(v)$  for all pairs of non-adjacent vertices  $u, v \in V(G)$ , and
  - (b)  $N_G[u] \triangle N_G[v]$  for all pairs of adjacent vertices  $u, v \in V(G)$ .

All results in this chapter also appear in [53] and [54].

### 7.1 Preliminary results

The following remark shows that in order to check if a total-dominating  $S$  of a graph  $G$  is an FTD-code of  $G$ , we do not need to check if  $S$  full-separates every pair of distinct vertices of  $G$  but only those which are at a distance of at most 2 between them.

**Remark 7.2.** *Let  $G$  be an FTD-admissible graph. A total-dominating set  $C$  of  $G$  is an FTD-code of  $G$  if and only if  $C$  full-separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \leq 2$ .*

*Proof.* The necessary condition for the statement follows immediately from the definition of an FTD-code. We, therefore, prove the sufficient condition. Thus, it is enough to show that  $C$  full-separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . So, assume  $u$  and  $v$  to be a pair of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . Since  $C$  is a total-dominating set of  $G$ , the vertex  $u$  has a neighbor, say  $w$ , in  $C$ . However,  $w$  is not a neighbor of  $v$  (since  $d_G(u, v) \geq 3$ ) and hence,  $w \in \Delta_F(G; u, v) \cap C$ , that is,  $w$  must be a separating  $C$ -codeword of the pair  $u, v$  in  $G$ . This proves the result.  $\square$

**Remark 7.3.** Let  $G$  be an FD-admissible graph and let  $C$  be an FD-code of  $G$ . Then, there exists at most one vertex  $w \in V(G)$  such that  $N_G(w) \cap C = \emptyset$ .

*Proof.* On the contrary, if there exist two distinct vertices  $u$  and  $v$  of  $G$  such that  $N_G(u) \cap C = N_G(v) \cap C = \emptyset$ , then it implies  $u$  and  $v$  are not open-separated and hence, also not full-separated, by  $C$ , contrary to our assumption. This proves the result.  $\square$

An equivalence of Remark 7.2 does not entirely apply for FD-codes. In other words, it is not true that any dominating set of a graph  $G$  that full-separates distinct vertices of  $G$  of distance at most 2 is an FD-code of  $G$ . As the next remark shows, for such a result to be true for FD-codes, we need the extra condition that the dominating set is “almost” a total-dominating set, that is, it total-dominates all vertices of  $G$  except possibly for one.

**Remark 7.4.** Let  $G$  be an FD-admissible graph and let  $C$  be a dominating set of  $G$  such that there exists at most one vertex  $w \in V(G)$  with  $N_G(w) \cap C = \emptyset$ . If the set  $C$  full-separates all distinct vertices  $u, v \in V(G)$  with  $d_G(u, v) \leq 2$ , then  $C$  is an FD-code of  $G$ .

*Proof.* It is enough to show that  $C$  full-separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \geq 3$ . In other words, since  $uv \notin E$ , it is enough to show that  $C$  open-separates  $u, v$ . Therefore, for two such vertices  $u, v \in V$  with  $d_G(u, v) \geq 3$ , we have  $N_G(u) \Delta N_G(v) = N_G(u) \cup N_G(v)$ . If  $C$  does not open-separate  $u$  and  $v$ , then it implies that  $(N_G(u) \cup N_G(v)) \cap C = (N_G(u) \Delta N_G(v)) \cap C = \emptyset$  and so,  $N_G(u) \cap C = N_G(v) \cap C = \emptyset$ . However, this contradicts our assumption. Therefore,  $C$  full-separates  $u$  and  $v$  and this proves the result.  $\square$

The next theorem demonstrates the closeness between an FD- and an FTD-number of a graph and hence, motivates us to compare the two numbers on graphs of some specific families to compare if they differ or are equal to each other on these graphs.

**Theorem 7.1** ([53]). Let  $G$  be an FD-admissible graph. If  $G$  is a disjoint union of a graph  $G'$  and an isolated vertex, then we have

$$\gamma^{\text{FD}}(G) = \gamma^{\text{FTD}}(G') + 1.$$

Otherwise, we have

$$\gamma^{\text{FTD}}(G) - 1 \leq \gamma^{\text{FD}}(G) \leq \gamma^{\text{FTD}}(G).$$

*Proof.* Let us first assume that  $G$  is a disjoint union of a graph  $G'$  and an isolated vertex  $v$ . Moreover, let  $C$  be a minimum FD-code of  $G$ . Then,  $v \in C$  in order for  $C$  to dominate  $v$ . This implies that  $N_G(v) \cap C = \emptyset$ . Therefore, by Remark 7.4, the set  $C' = C \setminus \{v\}$  must total-dominate all vertices in  $V \setminus \{v\}$ . Moreover, as  $v$  is an isolated vertex of  $G$ , the set  $C'$  is also a full-separating set of  $G'$  and hence, an FTD-code of  $G'$ . This implies that  $\gamma^{\text{FTD}}(G') \leq |C'| - 1 = \gamma^{\text{FD}}(G) - 1$ , that is,  $\gamma^{\text{FD}}(G) \geq \gamma^{\text{FTD}}(G') + 1$ . On the other hand, if  $C'$  is a minimum FTD-code of  $G'$ , then  $C' \cup \{v\}$  is an FD-code of  $G$ . This implies that  $\gamma^{\text{FD}}(G) \leq |C'| + 1 = \gamma^{\text{FTD}}(G') + 1$  and hence, the result follows.

Let us now assume that the graph  $G$  has no isolated vertices. Then,  $G$  is also FTD-admissible. Now, the inequality  $\gamma^{\text{FD}}(G) \leq \gamma^{\text{FTD}}(G)$  holds by Theorem 3.2. To prove the other inequality, we let  $C$  be a minimum FD-code of  $G$ . Then, we have  $|C| = \gamma^{\text{FD}}(G)$ . Now, if  $C$  is a total-dominating set of  $G$ , then  $C$  is also an FTD-code of  $G$  and hence, the result holds trivially. Therefore, let us assume that  $C$  is not a total-dominating set of  $G$ . This implies that there exists a vertex  $v$  such that  $N_G(v) \cap C = \emptyset$ . Moreover, Remark 7.3 implies that  $v$  is the only vertex of  $G$  whose neighborhood has an empty intersection with  $C$ . Since  $G$  has no isolated vertices by assumption, the vertex  $v$

has a neighbor, say  $u$ , in  $G$ . Then,  $C \cup \{u\}$  is a total-dominating set of  $G$  and hence, is also a full-separating set of  $G$ . Therefore, we have  $\gamma^{\text{FTD}}(G) \leq |C| + 1 = \gamma^{\text{FD}}(G) + 1$ .  $\square$

The second relation between the FD- and the FTD-number of a graph in Theorem 7.1 was also presented in Corollary 3.6 of Section 3.3 in relation to the comparisons of code-number pairs depicted in Figure 3.2 of the same section.

It can be verified that a vertex subset  $C$  of an FTD-admissible graph  $G$  is an FTD-code of  $G$  if and only if  $C \cap V(G')$  is an FTD-code of each component  $G'$  of  $G$ . As a result, the following remark shows that the FTD-number of an FTD-admissible graph is the same as the sum of the FTD-numbers of each component of  $G$ .

**Remark 7.5.** *For a positive integer  $\ell$ , let  $G_1, G_2, \dots, G_\ell$  be the components of an FTD-admissible graph  $G$ . Then,  $G_i$ , for each  $i \in [\ell]$ , is FTD-admissible and*

$$\gamma^{\text{FTD}}(G) = \sum_{i=1}^{\ell} \gamma^{\text{FTD}}(G_i).$$

As we shall see later (in Proposition 7.1), an equivalent of Remark 7.5 is not true for FD-numbers. Thus, along with OD-codes, FD-codes are the only ones (among the codes studied in this thesis) for which an equivalent of Remark 7.5 is not true.

## 7.2 FD- and FTD-numbers of some graph families

In this section, we study the FD- and the FTD-numbers of graphs belonging to some well-known graph families. Moreover, motivated by Theorem 7.1 where the FD-number and the FTD-number differ by at most 1, we compare the two numbers on graphs of these families. This comparison also exhibits extremal cases for the upper bounds in Theorem 3.4.

### 7.2.1 Paths and cycles

In order to study the FD- and FTD-numbers of paths  $P_n$  and cycles  $C_n$ , we first note that  $P_2$  and  $C_3$  have closed twins; and  $P_3$  and  $C_4$  have open twins. However, apart from them, all other paths and cycles are FD- and FTD-admissible. The proof of the following theorem giving the exact values of the FD- and FTD-numbers for paths and cycles is based on results by [190] and Theorem 6.4 (also in [26]), respectively.

**Lemma 7.1.** *Let  $G$  be either a path  $P_n$  for  $n \geq 4$  or a cycle  $C_n$  for  $n \geq 5$ . Then, we have  $\gamma^{\text{FD}}(G) = \gamma^{\text{FTD}}(G)$ .*

*Proof.* The graph  $G$  has maximum degree 2. On the contrary, let us assume that  $\gamma^{\text{FD}}(G) \neq \gamma^{\text{FTD}}(G)$ . Moreover, let  $C$  be a minimum FD-code of  $G$ . Therefore, we have  $|C| = \gamma^{\text{FD}}(G)$ . Therefore,  $C$  is not a total-dominating set of  $G$  (or else  $C$  would be an FTD-code of  $G$  thus implying  $\gamma^{\text{FD}}(G) = \gamma^{\text{FTD}}(G)$ , a contradiction). Now, by Remark 7.4, there exists exactly one vertex  $w$  of  $G$  such that  $N_G(w) \cap C = \emptyset$ . Thus,  $w \in C$  in order for  $C$  to dominate  $w$ . Let us first assume that  $w$  is not a full-separation forced vertex. Since  $N_G(w) \cap C = \emptyset$ , every vertex  $w' \in N_G(w)$  has a neighbor other than  $w$  in  $C$  (or else,  $w$  and  $w'$  are not full-separated by  $C$ , a contradiction). Therefore, for any neighbor  $w'$  of  $w$  in  $G$ , the set  $C' = (C \setminus \{w\}) \cup \{w'\}$  is a total-dominating set of  $G$ . Moreover, since  $w \in C$  is not full-separation forced, the set  $C'$  is also a full-separating set of  $G$  and hence, is an FTD-code of  $G$ . Moreover, we have  $|C'| = |C| = \gamma^{\text{FD}}(G)$ . This implies that  $\gamma^{\text{FTD}}(G) \leq \gamma^{\text{FD}}(G)$  and hence,  $\gamma^{\text{FD}}(G) = \gamma^{\text{FTD}}(G)$ , contrary to our assumption.

Therefore, let us assume that the vertex  $w$  is full-separation forced with respect to  $u$  and  $v$ . This implies, without loss of generality, that  $u \in N_G(w)$  and  $v \notin N_G[w]$ . Moreover, since  $N_G(w) \cap C = \emptyset$  and  $v \neq w$ , by Remark 7.3, we must have  $N_G(v) \cap C \neq \emptyset$ . Therefore, let  $x \in N_G(v) \cap C$ . Then,  $x \notin N_G[w]$ . Now, since  $C$  full-separates  $u$  and  $v$  uniquely by  $w$  (since  $w$  is full-separation forced), we must have  $x \in N_G(u)$  as well. Since  $u \notin C$ , in order for  $C$  to full-separate  $v$  and  $x$ , there must exist another vertex  $y \in (N_G(v) \setminus N_G(x)) \cap C$  (notice that  $y$  cannot be a neighbor of  $x$  since  $\{u, v\} \subset N_G(x)$  and  $\deg(x) \leq 2$ ). Moreover,  $uy \notin E$  since  $\{w, x\} \subset N_G(u)$  and  $\deg(u) \leq 2$ . Thus, the set  $C$  full-separates  $u$  and  $v$  by  $y$ , a contradiction to the uniqueness of  $w$ . Hence, we must have  $\gamma^{\text{FD}}(G) = \gamma^{\text{FTD}}(G)$ .  $\square$

**Observation 7.1.** *Let  $G = (V, E)$  be either a path  $P_n$  for  $n \geq 4$  or a cycle  $C_n$  for  $n \geq 5$ . Let  $C$  be an FTD-code of  $G$ . Then, for any vertex  $v \in V$ , there exists a vertex subset  $S_v \subset C$  of  $G$  such that  $v \in S_v$ ,  $|S_v| \geq 4$  and  $G[S_v]$  is the path  $P_{|S_v|}$ .*

*Proof.* Let  $V = \{v_1, v_2, \dots, v_n\}$  such that  $v_i v_{i+1} \in E$  for all  $i \in [n-1]$  and  $v_n v_1 \in E$  if  $G \cong C_n$ . Let  $v = v_i \in C$  for some  $i \in [n]$ . In order for  $C$  to total-dominate  $v_i$ , we must have  $v_{i+1} \in C$ , say, without loss of generality. Next, in order for  $C$  to full-separate  $v_i$  and  $v_{i+1}$ , we must have  $v_{i+2} \in C$ , say, without loss of generality. Again, in order for  $C$  to full-separate  $v_i$  and  $v_{i+2}$ , we must have  $v_{i+3} \in C$ , say, without loss of generality. Thus, the result follows by taking  $S = \{v_i, v_{i+1}, v_{i+2}, v_{i+3}\}$ .  $\square$

**Lemma 7.2.** *Let  $G$  be either a path  $P_n$  for  $n \geq 4$  or a cycle  $C_n$  for  $n \geq 5$ . Let  $n = 6q + r$ , where  $q$  and  $r$  are non-negative integers with  $r \in [0, 5]$ . Then we have*

$$\gamma^{\text{FTD}}(G) \leq \begin{cases} 4q + r, & \text{if } r \in [0, 4]; \\ 4q + 4, & \text{if } r = 5. \end{cases}$$

*Proof.* Let  $V = V(G) = \{v_1, v_2, \dots, v_n\}$  and that  $n = 6q + r$ , where  $r \in [0, 5]$ . If  $q = 0$ , the graph  $G$  is either a 4-path or a 5-path or a 5-cycle. In all the three cases, it can be checked that the set  $\{v_1, v_2, v_3, v_4\}$  is an FTD-code of  $G$ . Thus, in these cases, the result holds. For the rest of this proof, therefore, we assume that  $n \geq 6$ , that is,  $q \geq 1$ . We now construct a vertex subset  $C$  of  $G$  by including in  $C$  the vertices  $v_{6k-4}, v_{6k-3}, v_{6k-2}, v_{6k-1}$  for all  $k \in [q]$ , the vertices  $v_{6q}, v_{6q+1}, \dots, v_{6q+r-1}$  if  $r \in [4]$  and the vertices  $v_{6q+1}, v_{6q+2}, v_{6q+3}, v_{6q+4}$  if  $r = 5$  (see Figure 7.1 for an example with the code vertices marked in black). Notice that, by construction, the vertex subset  $C$  is a total-dominating set of  $G$ . Therefore, by Remark 7.2, to show that  $C$  is an FTD-code of  $G$ , it is enough to show that  $C$  full-separates every pair of distinct  $u, v \in V$  such that  $d_G(u, v) \leq 2$ . Now, one can observe that for every pair  $v_i, v_j$  of distinct vertices in  $G$  such that  $i < j \leq i + 2$  (that is  $d_G(v_i, v_j) \leq 2$ ), either  $v_{i-1}$  or  $v_{j+1}$  (or both) is in  $C$ . Thus, all vertex pairs  $v_i, v_j \in V$  with  $d_G(v_i, v_j) \leq 2$  are full-separated by  $C$ . This implies that  $C$  is an FTD-code of  $G$ . Hence, by counting the order of  $C$  from its construction, the result follows by  $\gamma^{\text{FTD}}(G) \leq |C|$ .  $\square$

**Theorem 7.2** ([54]). *Let  $G$  be either a path  $P_n$  for  $n \geq 4$  or a cycle  $C_n$  for  $n \geq 5$ . Moreover, let  $n = 6q + r$  for non-negative integers  $q$  and  $r \in [0, 5]$ . Then, we have*

$$\gamma^{\text{FD}}(G) = \gamma^{\text{FTD}}(G) = \begin{cases} 4q + r, & \text{if } r \in [0, 4]; \\ 4q + 4, & \text{if } r = 5. \end{cases}$$

*Proof.* Let  $V = V(G) = \{v_1, v_2, \dots, v_n\}$  such that  $v_i v_{i+1} \in E$  for all  $i \in [n-1]$  and  $v_n v_1 \in E$  if  $G \cong C_n$ . By Lemma 7.1, it is enough to prove the result only for  $\gamma^{\text{FTD}}(G)$ . To begin with, let us assume that  $G$  is a path  $P_n$  for  $n \geq 4$ . Then, by Theorem 3.2, we have

$$\gamma^{\text{FTD}}(P_n) \geq \gamma^{\text{OTD}}(P_n) = \begin{cases} 4q + r, & \text{if } r \in [0, 4] \\ 4q + 4, & \text{if } r = 5. \end{cases}$$

The last inequality is due to Seo and Slater [190]. Therefore, by Lemma 7.2, the result holds for the FTD-numbers of paths on  $n \geq 4$  vertices.

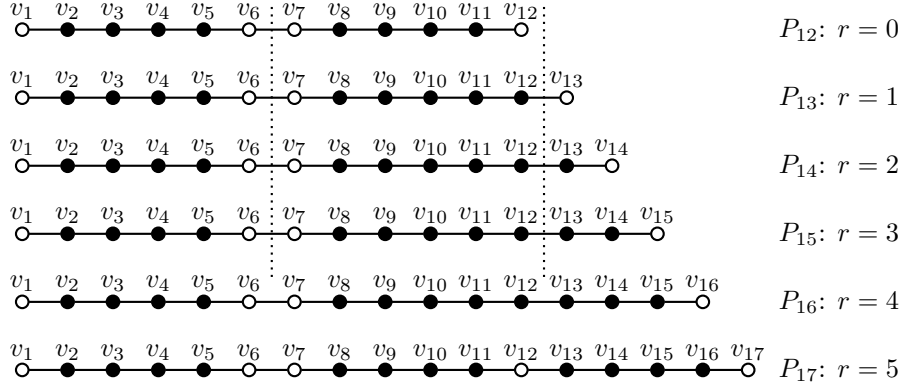


Figure 7.1: The set of black vertices represents an FTD-code of a path  $P_n$  with  $n \geq 4$  and a cycle  $C_n$  with  $n \geq 5$  (by joining the vertices  $v_1$  and  $v_n$  by an edge in each case).

We now prove the result for  $\gamma^{\text{FTD}}(G)$  when  $G$  is a cycle  $C_n$  for  $n \geq 5$ . By Theorem 6.4 (also in [26]), we have

$$\gamma^{\text{OTD}}(C_n) = \begin{cases} 4q + r, & \text{if } r = 0, 1, 2, 4 \\ 4q + 2, & \text{if } r = 3 \\ 4q + 4, & \text{if } r = 5. \end{cases} \quad (7.1)$$

Then again, by Lemma 7.2, the result for  $\gamma^{\text{FTD}}(C_n)$  holds for all  $n = 6q + r$  except for  $r = 3$ . Therefore, we next analyze only the  $(6q + 3)$ -cycles. So, let  $n = 6q + 3$  for any positive integer  $q$ . Using Lemma 7.2, we notice that it is enough to prove that  $\gamma^{\text{FTD}}(C_n) \geq 4q + 3$ . So, let us assume on the contrary that there exists an FTD-code  $C$  of  $G$  such that  $|C| \leq 4q + 2$ .

■ **Claim 1.** For some  $w \in V$ , there exists a vertex subset  $S_w \subset C$  of  $G$  (as in Observation 7.1) such that  $w \in S_u$ ,  $|S_w| \geq 5$  and  $G[S_w]$  is the path  $P_{|S_w|}$ .

*Proof of claim.* Without loss of generality, let us assume that, for every  $v \in V$ , the sets  $S_v$  as in Observation 7.1 are maximal. Contrary to the claim, let us assume that  $|S_v| \leq 4$  for all  $v \in V$ . Then, again using Observation 7.1, we have  $|S_v| = 4$ , for all  $v \in V$ . Define  $u \sim v$  if  $S_u = S_v$ . For some  $i < j$  and  $v_i \not\sim v_j$ , if we have  $\max\{i' : v_{i'} \in S_{v_i}\} = \min\{j' : v_{j'} \in S_{v_j}\}$ , then it implies that  $|S_{v_i}| \geq 8$ , a contradiction. Hence, for every  $i < j$  and  $v_i \not\sim v_j$ , there must be a vertex  $v_k \in V \setminus C$  such that  $\max\{i' : v_{i'} \in S_{v_i}\} < k < \min\{j' : v_{j'} \in S_{v_j}\}$ . This implies that  $|C| = 4q'$ , where  $q' \in [q]$ . Now, for any  $i < j$  and  $v_i \not\sim v_j$ , let  $v_k, v_{k+1}, \dots, v_{k+k'} \in V \setminus C$  such that  $\max\{i' : v_{i'} \in S_{v_i}\} < k < k+1 < \dots < k+k' < \min\{j' : v_{j'} \in S_{v_j}\}$ . Then we must have  $k' \in \{0, 1\}$ , or else, the vertex  $v_{k+1}$  is not total-dominated by  $C$ , a contradiction. Therefore, by counting, we must have  $n \leq 6q' \leq 6q$  which is a contradiction since  $n = 6q + 3$ . This proves our claim. ■

Now, let  $S = S_w$  and  $v_i, v_{i+1} \in S$ . Then, let  $G'$  be the  $(6q + 2)$ -cycle obtained by contracting the edge  $v_i v_{i+1}$ . Let  $V' = V(G') = \{u_1, u_2, \dots, u_{6q+2}\}$ , where  $u_j = v_j$  for  $j \in [i]$  and  $u_j = v_{j+1}$  for  $j \in [i+1, 6q+2]$ . Moreover, let  $C' = \{u_j : v_j \in C\}$ . Then  $C'$  is also a total-dominating set of  $G'$ . Moreover, we have  $|S_{u_j}| \geq 4$  for all  $j \in [6q+2]$ . For any two vertices  $u_j, u_{j'} \in V'$ , where  $j < j'$  and  $j' - j \leq 2$ , we can observe that either  $u_{j-1}$  or  $u_{j'+1}$  (or both) is in  $C'$ . Hence, by Remark 7.2, the set  $C'$  is an FTD-code of  $G'$ . However, this is a contradiction to Equation (7.1), since now we have  $\gamma^{\text{FTD}}(C_{6q+2}) \leq |C'| = 4q + 1$ . Hence,  $|C| \geq 4q + 3$  and this proves the result. □

## 7.2.2 Half-graphs

Let  $B_k = (U \cup W, E)$  be a half-graph with vertices in  $U = \{u_1, \dots, u_k\}$ ,  $W = \{w_1, \dots, w_k\}$  and edges  $u_i w_j$  if and only if  $i \leq j$ . In particular, we have  $B_1 = K_2$  and  $B_2 = P_4$ . Moreover, we clearly see that



half-graphs are connected and twin-free for all  $k \geq 2$  and hence, are both FD- and FTD-admissible in this case.

In [89] it was shown that the only graphs whose OTD-numbers equal the order of the graph are the disjoint unions of half-graphs. In particular, we have  $\gamma^{\text{OTD}}(B_k) = 2k$ . Combining this result with Theorem 3.2 showing that  $\gamma^{\text{OTD}}(B_k) \leq \gamma^{\text{FTD}}(B_k)$  and with Theorem 7.1 showing that  $\gamma^{\text{FD}}(G \oplus K_1) = \gamma^{\text{FTD}}(G) + 1$ , we have the following result.

**Corollary 7.1.** *For a graph  $G$  of order  $n$  being the disjoint union of half-graphs, we have  $\gamma^{\text{FTD}}(G) = n$  and  $\gamma^{\text{FD}}(G \oplus K_1) = \gamma^{\text{FTD}}(G) + 1 = n + 1$ .*

Hence, there are graphs where the FD- and FTD-numbers equal the order of the graph. A further example is  $B_2 = P_4$  with  $\gamma^{\text{FD}}(P_4) = 4$ . Conversely, we have:

**Theorem 7.3** ([54]). *For a half-graph  $B_k$  with  $k \geq 3$ , we have  $\gamma^{\text{FD}}(B_k) = 2k - 1$ .*

*Proof.* Let  $B_k = (U \cup W, E)$  be a half-graph with  $U = \{u_1, \dots, u_k\}$ ,  $W = \{w_1, \dots, w_k\}$  and edges  $u_i w_j$  if and only if  $i \leq j$ . In order to calculate  $\gamma^{\text{FD}}(B_k)$ , we first construct  $\mathcal{H}_{\text{FD}}(B_k) = \mathcal{H}_{\text{sep-F}}^1(B_k) \oplus \mathcal{H}_{\text{sep-F}}^{2+}(B_k) \oplus \mathcal{H}_{\text{dom-C}}(B_k)$ . Therefore, using Observation 2.5, the hyperedge set of  $\mathcal{H}_{\text{FD}}(B_k)$  consists of the closed neighborhoods

- $N_{B_k}[u_i] = \{u_i\} \cup \{w_i, \dots, w_k\}$  for all  $u_i \in U$ ,
- $N_{B_k}[w_j] = \{u_1, \dots, u_j\} \cup \{w_j\}$  for all  $w_j \in W$ ,

the symmetric differences of closed neighborhoods of adjacent vertices

- $N_{B_k}[u_i] \triangle N_{B_k}[w_j] = \{u_1, \dots, u_{i-1}, u_{i+1}, \dots, u_j\} \cup \{w_i, \dots, w_{j-1}, w_{j+1}, \dots, w_k\}$  for  $i < j$ ,
- $N_{B_k}[u_i] \triangle N_{B_k}[w_i] = \{u_1, \dots, u_{i-1}\} \cup \{w_{i+1}, \dots, w_k\}$

and the symmetric differences of open neighborhoods of non-adjacent vertices

- $N_{B_k}(u_i) \triangle N_{B_k}(u_j) = \{w_i, \dots, w_{j-1}\}$  for  $i < j$ ,
- $N_{B_k}(w_i) \triangle N_{B_k}(w_j) = \{u_{i+1}, \dots, u_j\}$  for  $i < j$ ,
- $N_{B_k}(u_i) \triangle N_{B_k}(w_j) = \{w_i, \dots, w_k\} \cup \{u_1, \dots, u_j\}$  for  $i > j$ .

In particular, we see that the following hold.

- $N_{B_k}(u_i) \triangle N_{B_k}(u_{i+1}) = \{w_i\}$  for  $1 \leq i < k$ ,
- $N_{B_k}(w_i) \triangle N_{B_k}(w_{i+1}) = \{u_{i+1}\}$  for  $1 \leq i < k$ , and
- $N_{B_k}(u_k) \triangle N_{B_k}(w_1) = \{w_k\} \cup \{u_1\}$ .

Therefore, for  $k \geq 3$ , all neighborhoods and all other symmetric differences are redundant hyperedges (note that this is not the case for  $k = 3$  as then  $N_{B_k}[u_1] \triangle N_{B_k}[w_1] = \{w_2\}$  and  $N_{B_k}[u_2] \triangle N_{B_k}[w_2] = \{u_1\}$  holds and makes  $N_{B_k}(u_2) \triangle N_{B_k}(w_1) = \{w_2\} \cup \{u_1\}$  redundant). This clearly shows that  $U \cup W \setminus \{w_k\}$  and  $U \setminus \{u_1\} \cup W$  are the only two minimum FD-codes of the half-graph  $B_k$  with  $k \geq 3$ , hence  $\gamma^{\text{FD}}(B_k) = 2k - 1$  indeed follows.  $\square$

Moreover, half-graphs are extremal graphs for the lower bounds  $\gamma^{\text{OTD}}(G) \leq \gamma^{\text{FTD}}(G)$  and  $\gamma^{\text{OTD}}(G) - 1 \leq \gamma^{\text{OD}}(G) \leq \gamma^{\text{FD}}(G)$  on graphs  $G$  (the latter is by combining Theorem 6.7 and Corollary 3.2).

**Proposition 7.1.** *For each positive integer  $\ell \geq 2$ , there exist connected FD-admissible graphs  $G_1, G_2, \dots, G_\ell$  such that*

$$\gamma^{\text{FD}}(G_1 \oplus G_2 \oplus \dots \oplus G_\ell) - \sum_{i=1}^{\ell} \gamma^{\text{FD}}(G_i) = \ell - 1.$$

*Proof.* Let  $G_1 \cong K_1$  and let  $G_2, \dots, G_\ell$  be graphs all isomorphic to the half-graph  $B_k$ . Then, we have  $\gamma^{\text{FD}}(G_1) = 1$  and, by Theorem 7.3, we have  $\gamma^{\text{FD}}(G_i) = 2k - 1$  for all  $i \in [2, \ell]$ . Thus,  $\sum_{i=1}^{\ell} \gamma^{\text{FD}}(G_i) = 1 + (2k - 1)(\ell - 1)$ . However, by Corollary 7.1, we have  $\gamma^{\text{FD}}(G_1 \oplus G_2 \oplus \dots \oplus G_\ell) = 1 + 2k(\ell - 1)$ . This implies the result.  $\square$

Thus, Proposition 7.1 shows that, along with OD-codes, it is not true, in general, for FD-codes that the FD-number of a graph is equal to the sum of the FD-numbers of its components.

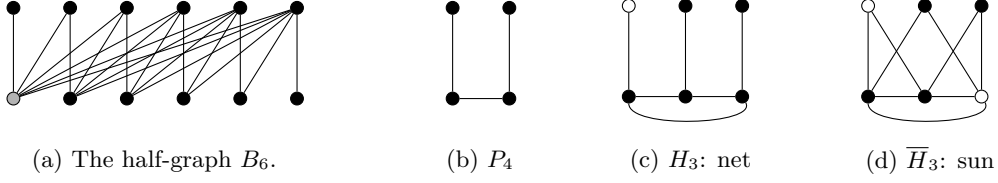


Figure 7.2: The black vertices in the figures represent minimum FD-codes. For Figures 7.2b, 7.2c and 7.2d, they also represent minimum FTD-codes. The gray vertex in Figures 7.2a depicts an extra vertex required to complete an FD- to the FTD-code.

### 7.2.3 Headless spiders

It is known from [11, 52] and Lemma 6.8 that  $\gamma^X(H_k) = k$  for  $X \in \{LD, LTD, OD, OTD\}$  and all  $k \geq 3$ . Furthermore,  $\gamma^{ID}(H_k) = k + 1$  for  $k \geq 3$  was shown in [6] and  $\gamma^{ITD}(H_k) = 2k - 1$  for  $k \geq 3$  in [101]. We next analyze the X-numbers of thin headless spiders for  $X \in \{FD, FTD\}$ .

**Theorem 7.4** ([54]). *For a thin headless spider  $H_k = (Q \cup S, E)$  with  $k \geq 4$ , we have  $\gamma^{FD}(H_k) = 2k - 2$  and  $\gamma^{FTD}(H_k) = 2k - 1$ .*

*Proof.* Let  $Q = \{q_1, \dots, q_k\}$  and  $S = \{s_1, \dots, s_k\}$ . In order to calculate  $\gamma^X(H_k)$  for  $X \in \{FD, FTD\}$ , we first construct the two X-hypergraphs  $\mathcal{H}_X(H_k) = \mathcal{H}_{\text{sep-F}}^1(G) \oplus \mathcal{H}_{\text{sep-F}}^{2+}(G) \oplus \mathcal{H}_{\text{dom-B}}(G)$ , where  $B \in \text{NBD-TYPE} = \{C, O\}$ . Therefore, the following hyperedges are involved: the closed or open neighborhoods, that is,

- $N_{H_k}[q_i] = Q \cup \{s_i\}$  or  $N_{H_k}(q_i) = Q \setminus \{q_i\} \cup \{s_i\}$  for all  $q_i \in Q$ ,
- $N_{H_k}[s_i] = \{q_i, s_i\}$  or  $N_{H_k}(s_i) = \{q_i\}$  for all  $s_i \in S$ ,

the symmetric differences of closed neighborhoods of adjacent vertices, that is,

- $N_{H_k}[s_i] \triangle N_{H_k}[q_i] = Q \setminus \{q_i\}$  for  $1 \leq i \leq k$ ,
- $N_{H_k}[q_i] \triangle N_{H_k}[q_j] = \{s_i, s_j\}$  for  $1 \leq i < j \leq k$ ,

and the symmetric differences of open neighborhoods of non-adjacent vertices, that is,

- $N_{H_k}(s_i) \triangle N_{H_k}(q_j) = Q \setminus \{q_i, q_j\} \cup \{s_j\}$  for  $i \neq j$ ,
- $N_{H_k}(s_i) \triangle N_{H_k}(s_j) = \{q_i, q_j\}$  for  $1 \leq i < j \leq k$ .

$\mathcal{H}_{FD}(H_k)$  is composed of the closed neighborhoods and all the symmetric differences. In particular, we see that

- $N_{H_k}[s_i] = \{q_i, s_i\}$  for  $1 \leq i \leq k$ ,
- $N_{H_k}[q_i] \triangle N_{H_k}[q_j] = \{s_i, s_j\}$  for  $1 \leq i < j \leq k$ ,
- $N_{H_k}(s_i) \triangle N_{H_k}(s_j) = \{q_i, q_j\}$  for  $1 \leq i < j \leq k$

belong to  $\mathcal{H}_{FD}(H_k)$ . Therefore,  $N_{H_k}[q_i]$  for all  $q_i \in Q$ ,  $N_{H_k}[s_i] \triangle N_{H_k}[q_i]$  for  $1 \leq i \leq k$ , and  $N_{H_k}(s_i) \triangle N_{H_k}(q_j)$  for  $i \neq j$  are redundant for  $k \geq 4$  (note that  $N_{H_k}(s_i) \triangle N_{H_k}(q_j)$  is not redundant if  $k = 3$ ). This shows that  $Q \setminus \{q_i\} \cup S \setminus \{s_j\}$  for  $i \neq j$  are the minimum FD-codes of  $H_k$ , hence  $\gamma^{FD}(H_k) = 2k - 2$  follows.

$\mathcal{H}_{FTD}(H_k)$  is composed of the open neighborhoods and all the symmetric differences. In particular, we have that

- $N_{H_k}(s_i) = \{q_i\}$  for  $1 \leq i \leq k$ ,
- $N_{H_k}[q_i] \triangle N_{H_k}[q_j] = \{s_i, s_j\}$  for  $1 \leq i < j \leq k$

belong to  $\mathcal{H}_{\text{FTD}}(H_k)$ . Therefore,  $N_{H_k}(q_i)$  for all  $q_i \in Q$ ,  $N_{H_k}[s_i] \triangle N_{H_k}[q_i]$  for  $1 \leq i \leq k$ ,  $N_{H_k}(s_i) \triangle N_{H_k}(q_j)$  for  $i \neq j$  and  $N_{H_k}(s_i) \triangle N_{H_k}(s_j)$  for  $1 \leq i < j \leq k$  are redundant (even for  $k \geq 3$ ). We conclude that  $Q \cup S \setminus \{s_i\}$  for  $1 \leq i \leq k$  are the minimum FTD-codes of  $H_k$  and  $\gamma^{\text{FTD}}(H_k) = 2k - 1$  holds accordingly.  $\square$

It is known from [11, 52] and Lemma 6.8 that  $\gamma^{\text{LD}}(\overline{H}_k) = \gamma^{\text{LTD}}(\overline{H}_k) = k - 1$  and  $\gamma^{\text{OD}}(\overline{H}_k) = \gamma^{\text{OTD}}(\overline{H}_k) = k + 1$  holds for all  $k \geq 3$ . Furthermore,  $\gamma^{\text{ID}}(\overline{H}_k) = k$  for  $k \geq 3$  was shown in [6]. We next determine the X-numbers of thick headless spiders for  $X \in \{\text{FD}, \text{FTD}, \text{ITD}\}$ .

**Theorem 7.5** ([54]). *For a thick headless spider  $\overline{H}_k = (Q \cup S, E)$  with  $k \geq 4$ , we have  $\gamma^{\text{ITD}}(\overline{H}_k) = k + 1$  and  $\gamma^{\text{FD}}(\overline{H}_k) = \gamma^{\text{FTD}}(\overline{H}_k) = 2k - 2$ .*

*Proof.* Let  $Q = \{q_1, \dots, q_k\}$  and  $S = \{s_1, \dots, s_k\}$ . In order to calculate  $\gamma^X(\overline{H}_k)$  for  $X \in \{\text{FD}, \text{FTD}, \text{ITD}\}$ , we first construct the three X-hypergraphs. Therefore, the following hyperedges are involved: the closed or open neighborhoods, that is,

- $N_{\overline{H}_k}[s_i] = Q \setminus \{q_i\} \cup \{s_i\}$  or  $N_{\overline{H}_k}(s_i) = Q \setminus \{q_i\}$  for all  $s_i \in S$ ,
- $N_{\overline{H}_k}[q_i] = Q \cup S \setminus \{s_i\}$  or  $N_{\overline{H}_k}(q_i) = Q \setminus \{q_i\} \cup S \setminus \{s_i\}$  for all  $q_i \in Q$ ,

the symmetric differences of closed neighborhoods of adjacent vertices, that is,

- $N_{\overline{H}_k}[s_i] \triangle N_{\overline{H}_k}[q_j] = \{q_i\} \cup S \setminus \{s_i, s_j\}$  for  $i \neq j$ ,
- $N_{\overline{H}_k}[q_i] \triangle N_{\overline{H}_k}[q_j] = \{s_i, s_j\}$  for  $i \neq j$ ,

and the symmetric differences of closed or open neighborhoods of non-adjacent vertices, that is,

- $N_{\overline{H}_k}[s_i] \triangle N_{\overline{H}_k}[q_i] = \{q_i\} \cup S$  or  $N_{\overline{H}_k}(s_i) \triangle N_{\overline{H}_k}(q_i) = S \setminus \{s_i\}$  for  $1 \leq i \leq k$ ,
- $N_{\overline{H}_k}[s_i] \triangle N_{\overline{H}_k}[s_j] = \{q_i, q_j\} \cup \{s_i, s_j\}$  or  $N_{\overline{H}_k}(s_i) \triangle N_{\overline{H}_k}(s_j) = \{q_i, q_j\}$  for  $i \neq j$ .

$\mathcal{H}_{\text{FD}}(\overline{H}_k)$  is composed of the closed neighborhoods and the symmetric differences of closed neighborhoods of adjacent vertices as well as of open neighborhoods of non-adjacent vertices. In particular, we see that

- $N_{\overline{H}_k}[q_i] \triangle N_{\overline{H}_k}[q_j] = \{s_i, s_j\}$  for  $1 \leq i < j \leq k$ ,
- $N_{\overline{H}_k}(s_i) \triangle N_{\overline{H}_k}(s_j) = \{q_i, q_j\}$  for  $1 \leq i < j \leq k$

belong to  $\mathcal{H}_{\text{FD}}(\overline{H}_k)$ . Therefore, all neighborhoods and the symmetric differences  $N_{\overline{H}_k}[s_i] \triangle N_{\overline{H}_k}[q_j]$  for  $i \neq j$  and  $N_{\overline{H}_k}(s_i) \triangle N_{\overline{H}_k}(q_i)$  for  $1 \leq i \leq k$  are redundant for  $k \geq 4$  (note that  $N_{\overline{H}_k}[s_i] \triangle N_{\overline{H}_k}[q_j]$  are not redundant if  $k = 3$ ). This shows that  $Q \setminus \{q_i\} \cup S \setminus \{s_j\}$  for  $1 \leq i, j \leq k$  are the minimum FD-codes of  $\overline{H}_k$ , hence  $\gamma^{\text{FD}}(\overline{H}_k) = 2k - 2$  follows.

$\mathcal{H}_{\text{FTD}}(\overline{H}_k)$  is composed of the open neighborhoods and the symmetric differences of closed neighborhoods of adjacent vertices as well as of open neighborhoods of non-adjacent vertices. Again, we have that

- $N_{\overline{H}_k}[q_i] \triangle N_{\overline{H}_k}[q_j] = \{s_i, s_j\}$  for  $1 \leq i < j \leq k$ ,
- $N_{\overline{H}_k}(s_i) \triangle N_{\overline{H}_k}(s_j) = \{q_i, q_j\}$  for  $1 \leq i < j \leq k$

belong to  $\mathcal{H}_{\text{FTD}}(\overline{H}_k)$ . Therefore, all neighborhoods and the other symmetric differences are redundant for  $k \geq 4$ . We conclude that  $Q \setminus \{q_i\} \cup S \setminus \{s_j\}$  for  $1 \leq i, j \leq k$  are also the minimum FTD-codes of  $\overline{H}_k$  and thus also  $\gamma^{\text{FTD}}(\overline{H}_k) = 2k - 2$  holds.

$\mathcal{H}_{\text{ITD}}(\overline{H}_k)$  is composed of the open neighborhoods and the symmetric differences of closed neighborhoods of distinct vertices. In particular, we see that

- $N_{\overline{H}_k}(s_i) = Q \setminus \{q_i\}$  for  $1 \leq i \leq k$ ,
- $N_{\overline{H}_k}[q_i] \triangle N_{\overline{H}_k}[q_j] = \{s_i, s_j\}$  for  $1 \leq i < j \leq k$

belong to  $\mathcal{H}_{\text{ITD}}(\overline{H}_k)$ . Thus,  $N_{\overline{H}_k}(q_i)$  for all  $q_i \in Q$  and all other symmetric differences are redundant for  $k \geq 4$  (note that  $N_{\overline{H}_k}[s_i] \triangle N_{\overline{H}_k}[q_j]$  are not redundant if  $k = 3$ ). This implies that two vertices from  $Q$  and all but one vertex from  $S$  have to be taken for every minimum ITD-code of  $\overline{H}_k$ , therefore  $\gamma^{\text{FTD}}(\overline{H}_k) = k + 1$  follows.  $\square$

Hence, for thick headless spiders, FD- and FTD-numbers are equal. This shows that thick headless spiders are further extremal cases for the lower bound  $\gamma^{\text{FD}}(G) \leq \gamma^{\text{FTD}}(G)$ , whereas thin headless spiders are extremal cases for the lower bounds  $\gamma^{\text{ITD}}(G) \leq \gamma^{\text{FTD}}(G)$  and  $\gamma^{\text{ITD}}(G) - 1 \leq \gamma^{\text{FD}}(G)$ .

## 7.3 Conclusion

In this chapter, we study the property of full separation in combination with both domination and total-domination. We addressed questions concerning the relation between the FD- and FTD-numbers thus showing that the two may differ by at most 1. This motivated us to compare the FD- and FTD-numbers on several graph families thus showing families where they are equal (for example, paths  $P_n$  with  $k \geq 4$ , cycles  $C_n$  with  $k \geq 5$ , thick headless spiders  $\overline{H}_k$  with  $k \geq 4$ ) or where they differ (for example, half-graphs  $B_k$  with  $k \geq 3$ , thin headless spiders  $H_k$  with  $k \geq 4$ ). In the process, we also provide examples of extremal cases for most of the lower bounds on FD- and FTD-numbers from Theorem 3.2.

Our lines of future research include to pursue these studies of the FD- and FTD-codes by involving more graph families, for example, to find more examples of graphs where the FD- or FTD-numbers are not close to the order of the graph. In general it would be interesting to identify or even characterize the extremal cases, that is, graphs for which the FD- or FTD-numbers equal the upper bound (the order of the graph) or one of its lower bounds in terms of other X-numbers.



## Part II

# Algorithmic aspects: Complexities and parameterized algorithms



# Chapter 8

## NP-hardness results

This chapter addresses the questions of computational complexities related to the problems of finding minimum OD-codes, FD-codes and FTD-codes. These three specific problems have been introduced to the literature of identification problems rather recently by Chakraborty (the author) and Wagler in [52] and [53] where the results in this chapter are also available.

Just as X-CODE is known to be NP-complete for all  $X \in \text{CODES} \setminus \{\text{OD}, \text{FD}, \text{FTD}\}$ , we show in this chapter that so are the problems of OD-CODE, FD-CODE and FTD-CODE (also see Tables 2.11 – 2.18 for a comprehensive overview of the complexities of the identification problems considered in this thesis). More formally, we show the following three decision problems to be NP-complete.

OD-CODE

**Input:** An OD-admissible graph  $G$  and a positive integer  $k$ .

**Question:** Does there exist an OD-code  $S$  of  $G$  such that  $|S| \leq k$ ?

FD-CODE

**Input:** An FD-admissible graph  $G$  and a positive integer  $k$ .

**Question:** Does there exist an FD-code  $S$  of  $G$  such that  $|S| \leq k$ ?

FTD-CODE

**Input:** An FTD-admissible graph  $G$  and a positive integer  $k$ .

**Question:** Does there exist an FTD-code  $S$  of  $G$  such that  $|S| \leq k$ ?

For each of the above three problems, we denote an instance by the 2-tuple  $(G, k)$ , where  $G$  is an X-admissible graph for  $X \in \{\text{OD}, \text{FD}, \text{FTD}\}$ , and  $k$  is a positive integer. In fact, to prove that the three problems are NP-complete, we can assume that  $k$  is at least the respective logarithmic lower bound given in Theorem 3.5. In other words, if the input graph to the above problems is on  $n$  vertices, we assume that for OD-CODE, FD-CODE and FTD-CODE, the integer  $k$  in their inputs is greater than  $\lceil \log n \rceil$ ,  $1 + \lfloor \log n \rfloor$  and  $1 + \lfloor \log(n+1) \rfloor$ , respectively. Otherwise, the respective problem can be answered with a NO in polynomial-time. On the other hand, if  $k \geq n$ , since the input graph  $G$  is X-admissible for each  $X \in \{\text{OD}, \text{FD}, \text{FTD}\}$ , it implies that the whole vertex set  $V(G)$  is an X-code of  $G$  of order at most  $k$ . Hence, in the case  $k \geq n$  as well, all three problems can be answered with YES in polynomial-time. As we shall see later, the above three problems will be NP-hard to decide for at least some integer  $k$  in between the logarithmic lower bound in Theorem 3.5 and  $n - 1$ .

We now recall from Theorems 6.7 and 7.1 the inequalities

$$\gamma^{\text{OTD}}(G) - 1 \leq \gamma^{\text{OD}}(G) \leq \gamma^{\text{OTD}}(G) \quad \text{and} \quad \gamma^{\text{FTD}}(G) - 1 \leq \gamma^{\text{FD}}(G) \leq \gamma^{\text{FTD}}(G),$$



respectively, for a graph  $G$  for which the appropriate codes exist. However, despite this small difference (of at most 1) between the respective pairs of code-numbers, we show that it is **NP**-hard to decide if they are equal or different. In other words, we show that it is **NP**-hard to decide if

$$\gamma^{\text{OTD}}(G) = \gamma^{\text{OD}}(G) \quad \text{or} \quad \gamma^{\text{OTD}}(G) = \gamma^{\text{OD}}(G) + 1$$

for an OTD-admissible graph  $G$ . Similarly, we show that it is **NP**-hard to decide if

$$\gamma^{\text{FTD}}(G) = \gamma^{\text{FD}}(G) \quad \text{or} \quad \gamma^{\text{FTD}}(G) = \gamma^{\text{FD}}(G) + 1$$

for an FTD-admissible graph  $G$ . This is equivalent to simply proving that it is **NP**-hard to decide if  $\gamma^{\text{OTD}}(G) = \gamma^{\text{OD}}(G) + 1$  and if  $\gamma^{\text{FTD}}(G) = \gamma^{\text{FD}}(G) + 1$ , respectively. More precisely, we prove the following two decision problems to be **NP**-hard (but not necessarily **NP**-complete).

**OD  $\neq$  OTD**

**Input:** An OTD-admissible graph  $G$  and an integer  $k$ .

**Question:** Is  $\gamma^{\text{OTD}}(G) = k$  and  $\gamma^{\text{OD}}(G) = k - 1$ ? That is, are the following assertions true?

- (a) There exists an OTD-code  $S$  of  $G$  such that  $|S| = k$ .
- (b) There exists an OD-code  $S'$  of  $G$  such that  $|S'| = k - 1$ .
- (c) For any vertex subset  $S''$  of  $G$ , if  $|S''| < |S|$ , then  $S''$  is not an OTD-code of  $G$ ; and if  $|S''| < |S'|$ , then  $S''$  is not an OD-code of  $G$ .

**FD  $\neq$  FTD**

**Input:** An FTD-admissible graph  $G$  and an integer  $k$ .

**Question:** Is  $\gamma^{\text{FTD}}(G) = k$  and  $\gamma^{\text{FD}}(G) = k - 1$ ? That is, are the following assertions true?

- (a) There exists an FTD-code  $S$  of  $G$  such that  $|S| = k$ .
- (b) There exists an FD-code  $S'$  of  $G$  such that  $|S'| = k - 1$ .
- (c) For any vertex subset  $S''$  of  $G$ , if  $|S''| < |S|$ , then  $S''$  is not an FTD-code of  $G$ ; and if  $|S''| < |S'|$ , then  $S''$  is not an FD-code of  $G$ .

As can be noticed in **OD  $\neq$  OTD**, given a vertex subset  $S$  (with  $|S| = k$ ) of an input graph  $G$  on  $n$  vertices, to check condition (c), one has to consider all subsets of  $V(G)$  of order at most  $k - 1$  (which admits  $\mathcal{O}(n^{k-1})$  as a worst-case running time) and check if  $S$  is an OTD-code (which admits  $\mathcal{O}(n^2)$  as a running time). In other words, verifying a certificate for **OD  $\neq$  OTD** can potentially take up to a running time of order  $\mathcal{O}(n^{k-1}) \cdot \mathcal{O}(n^2) \subseteq \mathcal{O}(n^{k+1})$  which may not be polynomial in  $n + k$ . Since we have  $|S| \in \mathcal{O}(n)$ , it implies that verifying a certificate for this problem may not necessarily be a polynomial-time algorithm. This implies that, **OD  $\neq$  OTD** does not necessarily belong to the class **NP**. For exactly similar reasons, **FD  $\neq$  FTD** does not necessarily belong to **NP** as well.

All our reductions in this chapter are from 3-SAT (or slight variations of it) and hence, we recall the problem here for reference.

**3-SAT**

**Input:** A set  $X$  of boolean variables and a set  $\mathcal{C}$  of clauses defined in terms of the literals from  $X$  such that each clause contains at most 3 literals.

**Question:** Does there exist a truth assignment on  $X$ ?

For 3-SAT, we call the 2-tuple  $(X, \mathcal{C})$  an instance of the problem, where  $X$  is a set of boolean variables and  $\mathcal{C}$  is a set of clauses with at most 3 literals from  $X$ . Moreover, for notational convenience, we denote an instance of the form  $(X, \mathcal{C})$  by  $\psi$ .

We also recall here Remarks 6.4, 7.2 and 7.4 which will come in handy for our proofs.

**Remark 6.4.** Let  $G$  be an OD-admissible graph and let  $C$  be a dominating set of  $G$  such that there exists at most one vertex  $w$  of  $G$  with  $N_G(w) \cap C = \emptyset$ . If the set  $C$  open-separates every pair  $(u, v)$  of distinct vertices of  $G$  with  $d_G(u, v) \leq 2$ , then  $C$  is an OD-code of  $G$ .

**Remark 7.2.** Let  $G$  be an FTD-admissible graph. A total-dominating set  $S$  of  $G$  is an FTD-code of  $G$  if and only if  $S$  full-separates every pair  $u, v$  of distinct vertices of  $G$  such that  $d_G(u, v) \leq 2$ .

**Remark 7.4.** Let  $G$  be an FD-admissible graph and let  $C$  be a dominating set of  $G$  such that there exists at most one vertex  $w \in V(G)$  with  $N_G(w) \cap C = \emptyset$ . If the set  $C$  full-separates all distinct vertices  $u, v \in V(G)$  with  $d_G(u, v) \leq 2$ , then  $C$  is an FD-code of  $G$ .

## 8.1 Open separation

In this section, we prove the following two hardness results (these results also appear in [52]).

**Theorem 8.1.** OD-CODE is NP-complete, even when the input graph is bipartite and of maximum degree at most 5.

**Theorem 8.2.**  $OD \neq OTD$  is NP-hard, even when the input graph is bipartite and of maximum degree at most 5.

To prove Theorems 8.1 and 8.2, we use LINEAR SAT, or LSAT — a variation of 3-SAT — defined in Section 2.1.4 in Chapter 2. We recall its definition below.

LINEAR SAT (LSAT)

**Input:** A set  $X$  of boolean variables and a set  $\mathcal{C}$  of clauses defined in terms of the literals from  $X$  such that each clause contains at most 3 literals; each literal from  $X$  can appear in at most two clauses; and any two distinct clauses can contain at most one literal in common.

**Question:** Does there exist a satisfying assignment on  $X$ ?

Also recall that LSAT was proven to be NP-complete by Arkin et al. in [12]. We now describe the following reduction from LSAT to both OD-CODE and  $OD \neq OTD$ .

**Reduction 8.1.** The reduction takes as input an instance  $\psi = (X, \mathcal{C})$  of LSAT, where  $X$  denotes the set of  $n$  variables and  $\mathcal{C}$  denotes the set of  $p$  clauses and let  $\phi$  be the boolean formula of LSAT with respect to  $X$  and  $\mathcal{C}$ . The reduction constructs a graph  $G^\psi$  as follows (also refer to Figure 8.1):

- The reduction runs over all literals from  $X$ . At each step, if a literal appears in only one clause, say  $\mathbf{c} \in \mathcal{C}$ , of  $\phi$  the reduction creates a dummy clause  $\mathbf{c}^d$  with exactly the same literals as in  $\mathbf{c}$  and updates the set  $\mathcal{C} \leftarrow \mathcal{C} \cup \{\mathbf{c}^d\}$  and appends the formula  $\phi \leftarrow \mathbf{c}^d \wedge \phi$  (notice that an assignment on  $X$  satisfies  $\phi$  if and only if it satisfies  $\phi \wedge \mathbf{c}^d$ ). At the end of this process, let  $m = |\mathcal{C}|$ . Then, we have  $m \leq p + n_1$ , where  $n_1$  is the number of literals from  $X$  that appear only once in  $\phi$ . This process makes sure that every literal in the formula  $\phi$  has at least two clauses (possibly dummy) in  $\mathcal{C}$  in which it appears.
- For every variable  $x \in X$ , do the following (refer to Figure 8.1a):
  - Add a vertex named  $w_1^x$  if the literal  $x$  appears in the formula  $\phi$ ; and add a vertex named  $w_2^x$  if the literal  $\neg x$  appears in the formula  $\phi$ . For the rest of the reduction, we assume that both  $w_1^x$  and  $w_2^x$  exist; and in case one of them does not, the corresponding action related to the vertex described in this reduction does not take place.
  - Add 3 vertices named  $v_1^x, v_2^x$  and  $v_3^x$  and add edges  $v_1^x w_1^x, v_1^x w_2^x$  and the edges  $v_1^x v_2^x$  and  $v_2^x v_3^x$  (the last two edges make the vertices  $v_1^x, v_2^x$  and  $v_3^x$  induce a  $P_3$ ).

Let the graph induced by the above 5 (or 4) vertices be denoted as  $G^x$  and be called the *variable gadget* corresponding to the variable  $x \in X$ .

- For every clause  $\mathbf{c} \in \mathcal{C}$ , do the following (refer to Figure 8.1b):

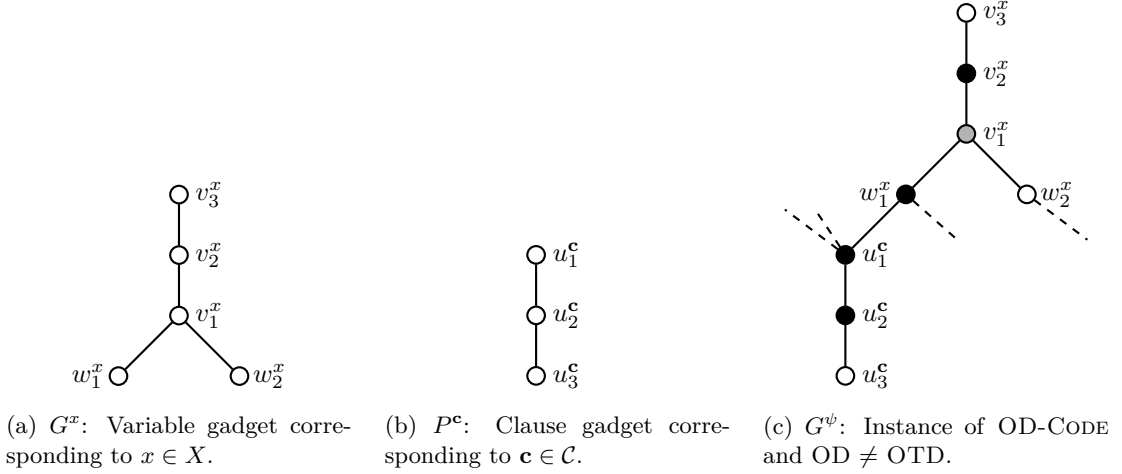


Figure 8.1: Polynomial-time construction of the graph  $G^\psi$  from an LSAT instance  $\psi = (X, \mathcal{C})$  as in Reduction 8.1. The black vertices in (c) represent those in a code described in Lemma 8.2. The gray vertex ( $v_1^x$ ) implies that, for some fixed variable  $x = x_0 \in X$ , the vertex is not included in an OD-code but is included in the OTD-code described in Lemma 8.2.

- Add 3 vertices named  $u_1^x$ ,  $u_2^x$  and  $u_3^x$  and add edges  $u_1^x u_2^x$  and  $u_2^x u_3^x$  thus making  $u_1^x$ ,  $u_2^x$  and  $u_3^x$  induce a  $P_3$ .

Let the  $P_3$  induced by the above 3 vertices be denoted by  $P^c$  and be called the *clause gadget* corresponding to the clause  $c \in \mathcal{C}$ .

- For all variables  $x \in X$  and all clauses  $c \in \mathcal{C}$ , if the literal  $x$  is in a clause  $c$ , then add the edge  $u_1^c w_1^x$ ; and if the literal  $\neg x$  is in a clause  $c$ , then add the edge  $u_1^c w_2^x$  (refer to Figure 8.1c).

As a matter of notation, for any variable  $x \in X$ , let us denote by  $[x]$  the set  $\{x, \neg x\}$ . Moreover, in all the proofs that follow in this section, we also assume that both the vertices  $w_1^x$  and  $w_2^x$ , for  $x \in X$ , exist in the graph  $G^\psi$ . In case one of them does not and the analysis changes as a result of that, we shall point that out accordingly.

**Lemma 8.1.** *For an instance  $\psi = (X, \mathcal{C})$  of LSAT with  $|X| = n$  and  $|\mathcal{C}| = m$ , let  $G^\psi$  be as in Reduction 8.1. Moreover, let  $T = \{(v_1^x, v_2^x, v_3^x) : x \in X\} \cup \{(u_1^c, u_2^c, u_3^c) : c \in \mathcal{C}\}$ . Then, for an open-separating set  $S$  of  $G^\psi$ , the following assertions are true.*

- (1) *If  $S$  is an OTD-code of  $G^\psi$ , then  $|T \cap S| \geq 2n + 2m$ .*
- (2) *If  $S$  is an OD-code of  $G^\psi$ , then  $|T \cap S| \geq 2n + 2m - 1$ .*

Moreover, we have  $\gamma^{\text{OTD}}(G^\psi) \geq 3n + 2m$  and  $\gamma^{\text{OD}}(G^\psi) \geq 3n + 2m - 1$ .

*Proof.* (1) Let  $S$  be an OTD-code of  $G^\psi$ . Then, for each  $(v_1, v_2, v_3) \in T$ , the vertex  $v_2$  must belong to  $S$  for the latter to total-dominate  $v_3$ . Moreover, at least one vertex from the pair  $(v_1, v_3)$  must be in  $S$  for the latter to total-dominate the vertex  $v_2$ . In total therefore, counting over all the  $n$  variable gadgets and  $m$  clause gadgets, we have  $|T \cap S| \geq 2n + 2m$ .

(2) Let  $S$  be an OD-code of  $G^\psi$ . Now, if at least two vertices out of each triple  $(t_1, t_2, t_3) \in T$  belong to  $S$ , then we have  $|S| \geq 2n + 2m > 2n + 2m - 1$  and thus, we are done. So, let us assume that there exists one triple  $(t_1^*, t_2^*, t_3^*) \in T$  which has at most one vertex in  $S$ . However, every triple  $(t_1, t_2, t_3) \in T$  must have at least one of  $t_2$  and  $t_3$  in  $S$  in order for the latter to dominate  $t_3$ . This implies that exactly one of  $t_2^*$  and  $t_3^*$  belongs to  $S$ . This further implies that either  $N_{G^\psi}(t_2^*) \cap S = \emptyset$  or  $N_{G^\psi}(t_3^*) \cap S = \emptyset$ . This further implies that each triple of  $T$  other than  $(t_1^*, t_2^*, t_3^*)$  has at least two vertices in  $S$  (or else, two vertices of  $G^\psi$  would have empty neighborhoods in  $S$  and hence, the pair

would not be open-separated by  $S$ , a contradiction). Thus, again counting over all the  $n$  variable gadgets and  $m$  clause gadgets, we have  $|S| \geq 2n + 2m - 1$ .

Finally, in each of the above two cases, since  $S$  is an open-separating set of  $G^\psi$ , at least one of  $w_1^x$  and  $w_2^x$  must be in  $S$  in order for the latter to open-separate the pair  $v_1^x, v_3^x$ . This implies the final statement of the result.  $\square$

**Lemma 8.2.** *For an instance  $\psi = (X, \mathcal{C})$  of LSAT with  $|X| = n$  and  $|\mathcal{C}| = m$ , let  $G^\psi$  be the graph as constructed in Reduction 8.1. Then, the existence of a satisfying assignment on  $X$  implies that*

- (1)  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$ ; and
- (2)  $\gamma^{\text{OTD}}(G^\psi) = 3n + 2m$ .

*Proof.* Let  $x_0 \in X$  be any fixed variable. We now look at the two cases.

(1) Since  $\gamma^{\text{OD}}(G^\psi) \geq 3n + 2m - 1$  by Lemma 8.1, in order to prove that  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$ , it is enough to show the existence of an OD-code of  $G^\psi$  of order at most  $3n + 2m - 1$ . Let us now construct a code  $S$  of  $G^\psi$  by including in it the following vertices.

- (a)  $v_1^x$  for all  $x \in X \setminus \{x_0\}$  and  $v_2^x$  for all  $x \in X$ .
- (b)  $u_1^c, u_2^c$  for all  $c \in \mathcal{C}$ .
- (c) For some  $x \in X$ , if exactly one of  $x$  and  $\neg x$  appears in the formula  $\phi$ , say  $x$ , then we pick  $w_1^x$  in  $S$ , or else, we pick  $w_2^x$  in  $S$ .
- (d) If for some variable  $x \in X$ , both the literals  $x$  and  $\neg x$  belong to some respective clauses, say  $c$  and  $c'$ , then exactly one of the literals is assigned 1 in the satisfying assignment on  $X$ . If  $x$  is assigned 1, we pick  $w_1^x$  in  $S$ , or else, we pick  $w_2^x$  in  $S$ .

This implies that every variable gadget has either  $w_1^x$  or  $w_2^x$  (but not both) in  $S$  and hence,  $|S| = 3n + 2m - 1$ . We now show that  $S$  is an OD-code of  $G^\psi$ . To start with, we observe that  $S$  is a dominating set of  $G^\psi$ . Thus, it is left to show that  $S$  is also an open-separating set of  $G$ . We also observe that for exactly the one vertex  $v_2^{x_0}$  of  $G^\psi$ , we have  $N_{G^\psi}(v_2^{x_0}) \cap S = \emptyset$ . Hence, to prove that  $S$ , indeed, is an open-separating set of  $G^\psi$ , by Remark 6.4, it is enough to check that all pairs of vertices of  $G^\psi$  of distance at most two between them are open-separated by  $S$ . We show next that this is true. Let  $x \in X$  and  $c \in \mathcal{C}$  be any general boolean variable and clause, respectively.

- Since  $N_{G^\psi}(v_2^{x_0}) \cap S = \emptyset$  uniquely, therefore,  $S$  open-separates  $v_2^{x_0}$  from every other vertex of  $G^\psi$ .
- The vertex  $v_2^x$  open-separates  $v_3^x$  from every vertex of  $G^\psi$  except  $v_1^x$ . However, the vertices  $v_1^x, v_3^x$  are open-separated by whichever of  $w_1^x$  and  $w_2^x$  is in  $S$ .
- Similarly, the vertex  $u_2^c$  open-separates  $u_3^c$  from every vertex of  $G^\psi$  except  $u_1^c$ . However, the clause  $c$  has a literal, say  $x$  or  $\neg x'$ , which is assigned 1 in the satisfying assignment on  $X$ . Then, the vertices  $u_1^c, u_3^c$  are open-separated by  $w_1^x$  if  $x$  is assigned 1 or by  $w_2^x$  if  $\neg x'$  is assigned 1.
- Every vertex of  $G^\psi$  in  $S$  open-separates itself from all its neighbors. Therefore, we now check that each vertex in  $S$  is also open-separated by  $S$  from all other vertices at distance 2 from the former.
- The vertex  $v_2^x$  open-separates  $v_1^x$  from every vertex of  $G^\psi$  except  $v_3^x$ . However, the pair  $v_1^x, v_3^x$  were previously shown to be open-separated by  $S$ .
- Similarly, the vertex  $u_2^c$  open-separates  $u_1^c$  from every vertex of  $G^\psi$  except  $u_3^c$ . Once again, the pair  $u_1^c, u_3^c$  were already shown to be open-separated by  $S$ .
- We now look at the open-separation of the pair  $v_2^x, w_i^x$  by  $S$ , where  $i \in [2]$ . Let us assume, without loss of generality, that the vertex  $w_1^x$  exists in the graph  $G^\psi$ . Then,  $w_1^x$  has a neighbor  $u_1^c$  for some clause  $c \in \mathcal{C}$ . Then, we have  $w_1^x u_1^c \in E(G^\psi)$ . Since  $u_1^c \in S$ , the pair  $v_2^x, w_1^x$  are open-separated by  $u_1^c \in S$ . The case for the open-separation of the pair  $v_2^x, w_2^x$  (if  $w_2^x$  exists in  $G^\psi$ ) by  $S$  follows by exactly the same reasoning.

- We now look at the open-separation by  $S$  of the vertex  $u_2^c$  from other vertices (of distance 2) from it. Let  $x \in X$  be such that  $x$  has a literal in  $\mathbf{c}$ . Without loss of generality, let  $x$  itself be a literal in  $\mathbf{c}$ . Thus, we look at the pair  $w_1^x, u_2^c$  to be open-separated by  $S$ . Since the literal  $x$  appears in two clauses, say  $\mathbf{c}$  and  $\mathbf{c}'$ , we have the edges  $w_1^x u_1^c \in E(G^\psi)$  and  $w_1^x u_1^{c'} \in E(G^\psi)$ . This implies that the vertices  $w_1^x, u_2^c$  are open-separated by  $u_1^c$ . Similarly, the vertices  $w_1^x, u_2^{c'}$  are open-separated by  $u_1^{c'}$  (this is where we use the property of the dummy clauses in the LSAT formula which makes sure that, for each  $x \in X$  and  $i \in [2]$ , the vertices  $w_i^x$  have two neighbors of the form  $u_1^c$  and  $u_1^{c'}$ , where  $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$ ).

Finally, the only vertices in different variable gadgets (respectively, clause gadgets) with distance 2 between them are of the form  $w_i^x \in G^x$  and  $w_j^{x'} \in G^{x'}$  (respectively,  $u_1^c \in P^c$  and  $u_1^{c'} \in P^{c'}$ ), where  $x, x' \in X$  (respectively,  $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$ ) are distinct. However, the set  $S$  open-separates any such pair  $w_i^x, w_j^{x'}$  by either  $v_1^x$  (if  $x_1 \neq x_0$ ) or by  $v_1^{x'}$  (if  $x'_1 \neq x'_0$ ); and the pair  $u_1^c, u_1^{c'}$  by  $u_2^c$ . This proves that  $S$  is an open-separating set of  $G^\psi$ .

(2) Again, by Lemma 8.1, we have  $\gamma^{\text{OTD}}(G^\psi) \geq 3n + 2m$ . Therefore, to show that  $\gamma^{\text{OTD}}(G^\psi) = 3n + 2m$ , it is enough to show that there exists an OTD-code of  $G^\psi$  of order at most  $3n + 2m$ . To that end, we simply replace the OD-code  $S$  constructed above in part (1) by  $S' = S \cup \{v_1^{x_0}\}$  which then becomes a total-dominating set of  $G^\psi$ . Moreover, since adding vertices to an open-separating set keeps it open-separating,  $S'$  is also an open-separating set of  $G^\psi$ . Moreover,  $|S'| = |S| + 1 = 3n + 2m$ . This proves the lemma.  $\square$

**Lemma 8.3.** *Let  $\psi = (X, \mathcal{C})$  be an instance of LSAT and  $G^\psi$  be the graph as in Reduction 8.1. If there exists an open-separating set  $S$  of  $G^\psi$  such that  $|\{w_1^x, w_2^x\} \cap S| = 1$  for all  $x \in X$ , then  $X$  has a satisfying assignment.*

*Proof.* Let  $S$  be an open-separating set of  $G^\psi$  such that  $|\{w_1^x, w_2^x\} \cap S| = 1$  for all  $x \in X$ . We now provide a binary assignment on  $X$  the following way: for any  $x \in X$ , if  $w_1^x \in S$  and  $w_2^x \notin S$ , then put  $(x, \neg x) = (1, 0)$ ; and if  $w_1^x \notin S$  and  $w_2^x \in S$ , then put  $(x, \neg x) = (0, 1)$ . Clearly, this assignment on  $X$  is a valid one since, for each  $x \in X$ , exactly one of  $x$  and  $\neg x$  is assigned 1 and the other 0. To now prove that this assignment on  $X$  is also a satisfying one, we simply note that, for each clause  $\mathbf{c} \in \mathcal{C}$ , in order for the set  $S$  to open-separate  $u_1^c$  and  $u_3^c$ , either there exists a variable  $x \in X$  such that  $u_1^c w_1^x \in E(G^\psi)$  with  $w_1^x \in S$ ; or there exists a variable  $x' \in X$  such that  $u_1^c w_2^{x'} \in E(G^\psi)$  and  $w_2^{x'} \in S$ . Therefore, by construction of the graph  $G^\psi$ , either  $x$  is a variable in the clause  $\mathbf{c}$  with  $(x, \neg x) = (1, 0)$  or  $x'$  is a variable whose literal  $\neg x'$  is in the clause  $\mathbf{c}$  with  $(x', \neg x') = (0, 1)$ . Hence, the binary assignment formulated on  $X$  is a satisfying one.  $\square$

**Lemma 8.4.** *For an instance  $\psi$  of LSAT with  $|X| = n$  and  $|\mathcal{C}| = m$ , let  $G^\psi$  be the graph as constructed in Reduction 8.1. Then, the following assertions are true.*

- (1) *If  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$ , then  $X$  has a satisfying assignment.*
- (2) *If  $\gamma^{\text{OTD}}(G^\psi) = 3n + 2m$ , then  $X$  has a satisfying assignment.*

*Proof.* Let  $S$  be either an OD-code or an OTD-code of  $G^\psi$ . Since  $S$  open-separates the vertices  $v_1^w, v_3^x$ , it implies that at least one of  $w_1^x$  and  $w_2^x$  must be in  $S$ . Then, by Lemma 8.3, to show that  $X$  has a valid satisfying assignment, it is enough to show that  $|\{w_1^x, w_2^x\} \cap S| = 1$ .

Let us first assume that  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$  and let  $S'$  be an OD-code of  $G^\psi$  such that  $|S'| = 3n + 2m - 1$ . If on the contrary, there exists some variable  $x' \in X$  for which  $|\{w_1^{x'}, w_2^{x'}\} \cap S'| = 2$ , then using Lemma 8.1(1), we have

$$3n + 2m - 1 = |S'| = |T \cap S'| + |\{w_1^{x'}, w_2^{x'}\} \cap S'| + \sum_{x \in X \setminus \{x'\}} |\{w_1^x, w_2^x\} \cap S'| \geq 3n + 2m.$$

which is a contradiction. Therefore,  $|\{w_1^{x'}, w_2^{x'}\} \cap S'| = 1$  and this proves (1).

Let us now assume that  $\gamma^{\text{OTD}}(G^\psi) = 3n + 2m$  and let  $S$  be an OTD-code of  $G^\psi$  such that  $|S| = 3n + 2m$ . Again, if on the contrary, there exists some variable  $x' \in X$  for which  $|\{w_1^{x'}, w_2^{x'}\} \cap S| = 2$ , then using Lemma 8.1(2), we have

$$3n + 2m = |S| = |T \cap S| + |\{w_1^{x'}, w_2^{x'}\} \cap S| + \sum_{x \in X \setminus \{x'\}} |\{w_1^x, w_2^x\} \cap S| \geq 3n + 2m + 1.$$

which is again a contradiction. Therefore,  $|\{w_1^{x'}, w_2^{x'}\} \cap S| = 1$  and this proves (2).  $\square$

**Lemma 8.5.** *For an instance  $\psi$  of LSAT, let  $G^\psi$  be the graph as constructed in Reduction 8.1. Then,  $G^\psi$  is bipartite and of maximum degree 5.*

*Proof.* Let  $V_1 = \{u_1^c, u_3^c, v_1^x, v_3^x : x \in X, c \in \mathcal{C}\}$  and  $V_2 = \{u_2^c, w_1^x, w_2^x, v_2^x : x \in X, c \in \mathcal{C}\}$ . Then it can be checked that the sets  $V_1$  and  $V_2$  are each independent sets. Moreover,  $V(G^\psi) = V_1 \cup V_2$ . This implies that  $G^\psi$  is a bipartite graph.

All other vertices in  $\{u_2^c, u_3^c, v_2^x, v_3^x\}$  have degree at most 3. Every clause  $c \in \mathcal{C}$  has at most 3 literals from  $X$ . This further implies that the vertex  $u_1^c$  is of degree at most 4. Starting from an instance of LSAT, at the end of the process of adding the dummy clauses in Reduction 8.1, it can be observed that any literal from  $X$  can be in at most four clauses<sup>1</sup>. This implies that the vertices  $w_1^x$  and  $w_2^x$  are of degree at most 5. This proves that the graph  $G^\psi$  is of maximum degree 5.  $\square$

This brings us to the proofs of our main theorems in this section.

**Theorem 8.1.** *OD-CODE is NP-complete, even when the input graph is bipartite and of maximum degree at most 5.*

*Proof.* OD-CODE clearly belongs to the class NP since it can be verified in polynomial-time if a given vertex subset of a graph  $G$  is an OD-code of  $G$ . To prove NP-hardness, we show that an instance  $\psi = (X, \mathcal{C})$  with  $|X| = n$  and  $|\mathcal{C}| = m$  is a YES-instance of LSAT if and only if  $(G^\psi, 3n + 2m - 1)$  is a YES-instance of OD-CODE, where the graph  $G^\psi$  is as constructed in Reduction 8.1. In other words, we show that  $X$  has a satisfying assignment if and only if there exists an OD-code  $S$  of  $G^\psi$  such that  $|S| \leq 3n + 2m - 1$ .

To prove the necessary part of the last statement, if  $X$  has a satisfying assignment, then by Lemma 8.2(1), we have  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$ . This implies that there exists an OD-code  $S$  of  $G^\psi$  such that  $|S| = 3n + 2m - 1$ . On the other hand, to prove the sufficiency, if there exists an OD-code  $S$  of  $G^\psi$  such that  $|S| \leq 3n + 2m - 1$ , then combining with Lemma 8.1, we have  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$ . Therefore, by Lemma 8.4(1), there exists a satisfying assignment on  $X$ .

Finally, by Lemma 8.5, the graph  $G^\psi$  is bipartite and of maximum degree 5. This proves the result.  $\square$

**Theorem 8.2.** *OD  $\neq$  OTD is NP-hard, even when the input graph is bipartite and of maximum degree at most 5.*

*Proof.* We show that OD  $\neq$  OTD is NP-hard by showing that an instance  $\psi = (X, \mathcal{C})$  with  $|X| = n$  and  $|\mathcal{C}| = m$  is a YES-instance of LSAT if and only if  $(G^\psi, 3n + 2m)$  is a YES-instance of OD  $\neq$  OTD, where the graph  $G^\psi$  is as constructed in Reduction 8.1. In other words, we show that  $X$  has a satisfying assignment if and only if  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$  and  $\gamma^{\text{OTD}}(G^\psi) = 3n + 2m$ .

The necessary condition is true since, if  $X$  has a satisfying assignment, then by Lemma 8.2, we have  $\gamma^{\text{OTD}}(G^\psi) = 3n + 2m$  and  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$ . To prove the sufficiency condition, let us assume that  $\gamma^{\text{OTD}}(G^\psi) = 3n + 2m$  and  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m - 1$ . Then, by Lemma 8.4,  $X$  has a satisfying assignment.

Finally, by Lemma 8.5, the graph  $G^\psi$  is bipartite and of maximum degree 5. This proves the result.  $\square$

<sup>1</sup>This could not be said if we reduced the problem from 3-SAT

Reduction 8.1 also proves OTD-CODE to be NP-complete. Even though this result was also shown in [190] by Seo and Slater, we believe that our reduction is perhaps a bit simpler than the one used in [190]. This actually answers the very same question posed by Slater [105] regarding the existence of an easier proof for the NP-completeness of OTD-CODE. Moreover, our result proves the OTD-CODE to be NP-complete even for bipartite graphs of maximum degree 5 (however, note that the latter result is weaker than the one in [172] where the authors show the problem to be NP-complete on bipartite graphs of maximum degree 4). In any case, we also present here the following formal proof.

**Theorem 8.3.** *OTD-CODE is NP-complete even when the input graph is bipartite and of maximum degree at most 5.*

*Proof.* OTD-CODE clearly belongs to the class NP since it can be verified in polynomial-time if a given vertex subset of a graph  $G$  is an OTD-code of  $G$ . To prove NP-hardness, we show that an instance  $\psi = (X, \mathcal{C})$  with  $|X| = n$  and  $|\mathcal{C}| = m$  is a YES-instance of LSAT if and only if  $(G^\psi, 3n+2m)$  is a YES-instance of OTD-CODE, where the graph  $G^\psi$  is as constructed in Reduction 8.1. In other words, we show that  $X$  has a satisfying assignment if and only if there exists an OTD-code  $S$  of  $G^\psi$  such that  $|S| \leq 3n + 2m$ .

To prove the necessary part of the last statement, if  $X$  has a satisfying assignment, then by Lemma 8.2(2), we have  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m$ . This implies that there exists an OTD-code  $S$  of  $G^\psi$  such that  $|S| = 3n + 2m$ . On the other hand, to prove the sufficiency, if there exists an OTD-code  $S$  of  $G^\psi$  such that  $|S| \leq 3n + 2m$ , then combining with Lemma 8.1, we have  $\gamma^{\text{OD}}(G^\psi) = 3n + 2m$ . Therefore, by Lemma 8.4(2), there exists a satisfying assignment on  $X$ .

Finally, by Lemma 8.5, the graph  $G^\psi$  is bipartite and of maximum degree 5. This proves the result.  $\square$

## 8.2 Full separation

In this section, we prove the following three hardness results (these results also appear in [53]).

**Theorem 8.4.** *FTD-CODE is NP-complete.*

**Theorem 8.5.** *FD-CODE is NP-complete.*

**Theorem 8.6.**  *$FD \neq FTD$  is NP-hard.*

To prove Theorems 8.4, 8.5 and 8.6, we use the following (single) reduction from an instance  $\psi = (X, \mathcal{C})$  of 3-SAT to an instance  $(G^\psi, k)$  of FD-CODE, FTD-CODE and  $FD \neq FTD$ .

**Reduction 8.2.** The reduction takes as input an instance  $\psi = (X, \mathcal{C})$  of 3-SAT. Let  $n = |X|$  and  $m = |\mathcal{C}|$ . The reduction constructs a graph  $G^\psi$  on  $10n + 3m$  vertices as follows (also refer to Figure 8.2):

- For every variable  $x \in X$ , do the following (refer to Figure 8.2(a)):
  - Add 10 vertices named  $v_1^x, v_2^x, v_3^x, w_1^x, w_2^x, s_1^x, s_2^x, s_3^x, z_1^x$  and  $z_2^x$  (intuitively, the vertex  $w_1^x$  will correspond to the literal  $x$  and the vertex  $w_2^x$  will correspond to the literal  $\neg x$ ).
  - Add edges so that the graph induced by the vertices  $v_3^x, w_1^x, w_2^x, s_1^x, s_2^x$  and  $s_3^x$  is a  $K_6$  minus the edges  $w_1^x w_2^x, v_1^x s_1^x, v_1^x s_2^x$  and  $v_1^x s_3^x$ .
  - Add edges  $v_1^x v_2^x$  and  $v_2^x v_3^x$  making the vertices  $v_1^x, v_2^x$  and  $v_3^x$  induce a  $P_3$ ; and add edges  $s_1^x z_1^x$  and  $s_2^x z_2^x$  making  $z_1^x$  and  $z_2^x$  pendant vertices and the vertices  $s_1^x$  and  $s_2^x$  their respective supports.

Let the graph induced by the above 10 vertices be denoted as  $G^x$  and be called the *variable gadget* corresponding to the variable  $x \in X$ .

- For every clause  $c \in \mathcal{C}$ , do the following (refer to Figure 8.2(b)):

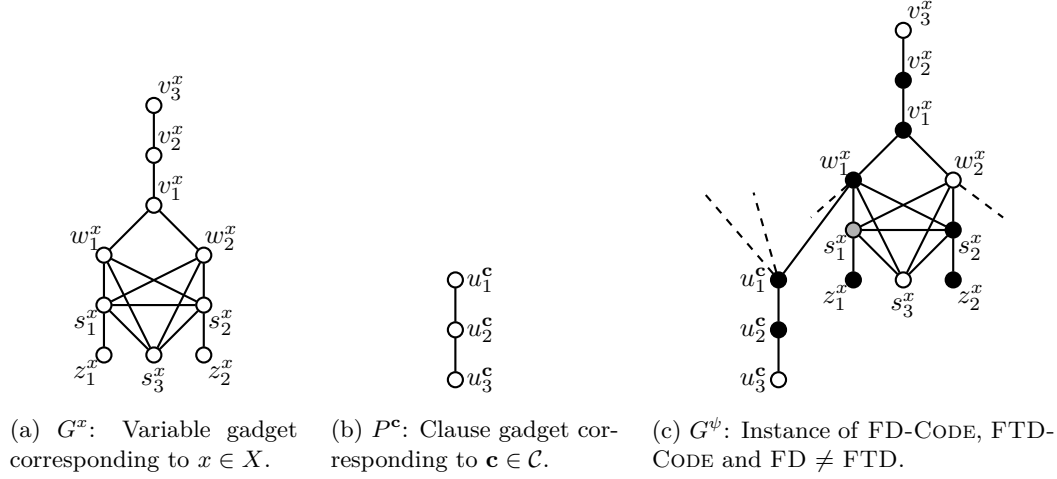


Figure 8.2: Polynomial-time construction of the graph  $G^\psi$  from an instance  $\psi = (X, \mathcal{C})$  of 3-SAT as in Reduction 8.2. The black vertices in (c) represent those in a code described in Lemma 8.7. The grey vertex ( $s_1^x$ ) implies that, for some fixed variable  $x = x' \in X$ , the vertex is not included in an FD-code but is included in the FTD-code described in Lemma 8.7.

- Add 3 vertices named  $u_1^c$ ,  $u_2^c$  and  $u_3^c$ .
- Add edges  $u_1^c u_2^c$  and  $u_2^c u_3^c$  thus making  $u_1^c$ ,  $u_2^c$  and  $u_3^c$  induce a  $P_3$ .

Let the  $P_3$  induced by the above 3 vertices be denoted by  $P^c$  and be called the *clause gadget* corresponding to the clause  $\mathbf{c} \in \mathcal{C}$ .

- For all variables  $x \in X$  and all clauses  $\mathbf{c} \in \mathcal{C}$ , if the literal  $x$  is in a clause  $\mathbf{c}$ , then add the edge  $u_1^c w_1^x$ ; and if the literal  $\neg x$  is in a clause  $\mathbf{c}$ , then add the edge  $u_1^c w_2^x$  (refer to Figure 8.2(c)).

It can be verified that Reduction 8.2 is carried out in time polynomial in  $m$  and  $n$ . The proofs of Theorems 8.4, 8.5 and 8.6 comprise of first proving the following lemmas.

**Lemma 8.6.** *Let  $S$  be a full-separating set of  $G^\psi$ .*

- (a) *If  $S$  is either an FD- or an FTD-code of  $G^\psi$ , then for all  $\mathbf{c} \in \mathcal{C}$ , we have  $|V(P^c) \cap S| \geq 2$ .*
- (b) *If  $S$  is an FTD-code, then we have  $|(V(G^x) \setminus \{w_1^x, w_2^x\}) \cap S| \geq 6$  for all  $x \in X$ .*
- (c) *If  $S$  is an FD-code, then except possibly for one  $x' \in X$  for which  $|(V(G^{x'}) \setminus \{w_1^{x'}, w_2^{x'}\}) \cap S| \geq 5$ , we have  $|(V(G^x) \setminus \{w_1^x, w_2^x\}) \cap S| \geq 6$  for all  $x \in X \setminus \{x'\}$ .*

Moreover, we have  $\gamma^{\text{FTD}}(G^\psi) \geq 7n + 2m$  and  $\gamma^{\text{FD}}(G^\psi) \geq 7n + 2m - 1$ .

*Proof.* (a): Let  $S$  be either an FD- or an FTD-code of  $G^\psi$  and let  $\mathbf{c} \in \mathcal{C}$ . Since  $u_3^c$  is a pendant vertex of  $G^\psi$ , we must have  $|\{u_2^c, u_3^c\} \cap S| \geq 1$ . Moreover, the vertex  $u_1^c$  is full-separation forced with respect to the pair  $u_2^c, u_3^c$ . Therefore, we also have  $u_1^c \in S$ . This proves (a).

(b) and (c): Let  $S$  be either an FD- or an FTD-code of  $G^\psi$  and let  $x \in X$ . Since  $v_3^x$  is also a pendant vertex of  $G^\psi$ , we must have  $|\{v_2^x, v_3^x\} \cap S| \geq 1$ . Since the vertex  $v_1^x$  is full-separation forced with respect to the pair  $v_2^x, v_3^x$  and, for  $i \in [2]$ , the vertex  $z_i^x$  is full-separation forced with respect to  $s_i^x, s_3^x$ , we must have  $v_1^x, z_1^x, z_2^x \in S$  for all  $x \in X$ . Furthermore, we have the following.

- If  $S$  is an FTD-code, for each  $i \in [2]$  and each  $x \in X$ , the vertex  $s_i^x$  is a support vertex of  $G^\psi$  and hence, is total-domination forced. Therefore, we have  $s_i^x \in S$ .
- If  $S$  is an FD-code, by Remark 7.3, at least all but one, that is,  $(n - 1)$  many, vertices in the set  $\{s_i^x \in V(G^\psi) : x \in X, i \in [2]\}$  must belong to  $S$ .



This proves parts (b) and (c).

Finally, we see that for all  $x \in X$ , in order for  $S$  to full-separate the pair  $v_1^x, v_3^x$ , we must have  $|\{w_1^x, w_2^x\} \cap S| \geq 1$ . Hence, by counting the vertices in  $S$ , the final statement of the result follows.  $\square$

**Lemma 8.7.** *If  $X$  has a satisfying assignment, then we have  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$  and  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$ .*

*Proof.* First, we show that there exists an FTD-code  $S$  of  $G^\psi$  such that  $|S| = 7n + 2m$ . To that end, we build a vertex subset  $S$  of  $G^\psi$  as follows.

- For all  $x \in X$  and  $\mathbf{c} \in \mathcal{C}$ , pick the vertices  $u_1^{\mathbf{c}}, u_2^{\mathbf{c}}, v_1^x, v_2^x, z_1^x, z_2^x, s_1^x, s_2^x$  in  $S$ .
- For any variable  $x \in X$ , either  $x$  or  $\neg x$  is assigned 1 in the satisfying assignment on  $X$ . If  $x$  is assigned 1, then pick  $w_1^x$  in  $S$ ; and if  $\neg x$  is assigned 1, then pick  $w_2^x$  in  $S$ .

Note that, by construction, we have  $|S| = 7n + 2m$ . We now show that  $S$  is an FTD-code of  $G^\psi$ . Notice that, again by construction, the set  $S$  is a total-dominating set of  $G^\psi$ . Therefore, by Remark 7.2, it is enough to show that  $S$  full-separates all pairs of distinct vertices of  $G^\psi$  with distance at most 2 between them. For every  $x \in X$  and  $\mathbf{c} \in \mathcal{C}$ , the set  $S$  full-separates

- the vertices  $v_2^x, v_3^x$  by  $v_1^x$ ; the vertices  $u_2^{\mathbf{c}}, u_3^{\mathbf{c}}$  by  $u_1^{\mathbf{c}}$ ; and the vertices  $v_1^x, u_1^{\mathbf{c}}$  by  $u_2^{\mathbf{c}}$ .
- It full-separates the vertices  $v_1^x, v_3^x$ , by either  $w_1^x$  or  $w_2^x$  according to whether  $(x, \neg x) = (1, 0)$  or  $(x, \neg x) = (0, 1)$ , respectively.
- It full-separates the vertices  $u_1^{\mathbf{c}}, u_3^{\mathbf{c}}$  by either  $w_1^x$  or  $w_2^{x'}$  for some  $x, x' \in X$  according to whether  $x$  (assigned to 1) belongs to the clause  $\mathbf{c}$ ; or  $\neg x'$  (assigned to 1) belongs to  $\mathbf{c}$ .
- It full-separates the vertices  $w_1^x, w_2^x$  by  $u_1^{\mathbf{c}}$  for some clause  $\mathbf{c} \in \mathcal{C}$  (to which exactly one of  $x$  and  $\neg x$  belongs)
- For any  $i \in [2]$  and  $j \in [3]$ , the set  $S$  full-separates the vertices  $w_i^x, s_j^x$  by  $v_1^x$ .
- For any  $i, j \in [2]$ , the set  $S$  full-separates the vertices  $w_i^x, z_j^x$  by  $v_1^x$ ; the vertices  $w_i^x, u_j^{\mathbf{c}}$  by  $v_1^x$ ; and the vertices  $w_i^x, v_j^{\mathbf{c}}$  by  $s_2^x$ .
- For any  $i, j \in [3]$ , it full-separates the vertices  $s_i^x, s_j^x$ , by either  $z_1^x$  or  $z_2^x$ .
- For any  $i \in [3]$ , it full-separates the vertices  $s_i^x, v_1^x$ , by  $z_i^x$ ; and the vertices  $s_i^x, u_1^{\mathbf{c}}$ , by  $z_i^x$ .
- $S$  full-separates the vertices  $s_i^x, z_j^x$ , by either  $w_1^x$  or  $w_2^x$  according to whether  $(x, \neg x) = (1, 0)$  or  $(x, \neg x) = (0, 1)$ , respectively, where  $i, j \in [2]$ ;

Finally, the only vertices in different variable gadgets (respectively, clause gadgets) with distance 2 between them are of the form  $w_i^x \in G^x$  and  $w_j^{x'} \in G^{x'}$  (respectively,  $u_1^{\mathbf{c}} \in P^{\mathbf{c}}$  and  $u_1^{\mathbf{c}'}$  in  $P^{\mathbf{c}'}$ ), where  $x, x' \in X$  (respectively,  $\mathbf{c}, \mathbf{c}' \in \mathcal{C}$ ) are distinct. However, the set  $S$  full-separates any such pair  $w_i^x, w_j^{x'}$  by  $v_1^x$  and the pair  $u_1^{\mathbf{c}}, u_1^{\mathbf{c}'}$  by  $u_2^{\mathbf{c}}$ . This proves that  $S$  is a full-separating set of  $G^\psi$ .

We now let  $S' = S \setminus \{s_1^{x'}\}$ , where  $x'$  is a fixed variable in  $X$ . Therefore, we have  $|S'| = 7n + 2m - 1$ . Hence, it is enough to show that  $S'$  is an FD-code of  $G^\psi$ . First of all, it can be verified that  $S'$  is a dominating set of  $G^\psi$ . We now show that  $S'$  is also a full-separating set of  $G^\psi$ . To do so, notice that  $N_{G^\psi}(z_1^{x'}) \cap S' = \emptyset$  for only the variable  $x' \in X$ . Therefore, by Remark 7.4, it is enough to show that  $S'$  full-separates all pairs of distinct vertices of  $G^\psi$  with distance at most 2 between them. In other words, we consider for full-separation by  $S'$  the exact same pairs of vertices we had considered for full-separation by  $S$  in the previous list. Moreover, we notice that in the previous list,  $S$  does not full-separate any pair of vertices of  $G^\psi$  by the vertex  $s_1^x$  for any  $x \in X$ . This implies that each of those pairs of vertices of  $G^\psi$  is also full-separated by  $S'$  as well and hence, this proves that  $S'$  too is a full-separating set of  $G^\psi$ .

Thus, the existence of the above FTD-code  $S$  and the FD-code  $S'$  implies that  $\gamma^{\text{FTD}}(G^\psi) \leq |S| = 7n + 2m$  and  $\gamma^{\text{FD}}(G^\psi) \leq |S'| = 7n + 2m - 1$ . Therefore, combining these with the last part of Lemma 8.6, the result follows.  $\square$

**Lemma 8.8.** *Let  $\psi = (X, \mathcal{C})$  be an instance of 3-SAT and  $G^\psi$  be the graph as in Reduction 8.2. If there exists a full-separating set  $S$  of  $G^\psi$  such that  $|\{w_1^x, w_2^x\} \cap S| = 1$  for all  $x \in X$ , then  $X$  has a satisfying assignment.*

*Proof.* Let  $S$  be a full-separating set of  $G^\psi$  such that  $|\{w_1^x, w_2^x\} \cap S| = 1$  for all  $x \in X$ . We now provide a boolean assignment on  $X$  the following way: for any  $x \in X$ , if  $w_1^x \in S$  and  $w_2^x \notin S$ , then put  $(x, \neg x) = (1, 0)$ ; and if  $w_1^x \notin S$  and  $w_2^x \in S$ , then put  $(x, \neg x) = (0, 1)$ . Clearly, this assignment on  $X$  is a valid one since, for each  $x \in X$ , exactly one of  $x$  and  $\neg x$  is assigned 1 and the other 0. To now prove that this assignment on  $X$  is also a satisfying one, we simply note that, for each clause  $\mathbf{c} \in \mathcal{C}$ , in order for the set  $S$  to full-separate  $u_1^{\mathbf{c}}$  and  $u_3^{\mathbf{c}}$ , either there exists a variable  $x \in X$  such that  $u_1^{\mathbf{c}}w_1^x \in E(G^\psi)$  with  $w_1^x \in S$ ; or there exists a variable  $x' \in X$  such that  $u_1^{\mathbf{c}}w_2^{x'} \in E(G^\psi)$  and  $w_2^{x'} \in S$ . Therefore, by construction of the graph  $G^\psi$ , either  $x$  is a variable in the clause  $\mathbf{c}$  with  $(x, \neg x) = (1, 0)$  or  $x'$  is a variable whose literal  $\neg x'$  is in the clause  $\mathbf{c}$  with  $(x', \neg x') = (0, 1)$ . Hence, the binary assignment formulated on  $X$  is a satisfying one.  $\square$

**Lemma 8.9.** *The following statements are true.*

- (1) *If  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$ , then  $X$  has a satisfying assignment.*
- (2) *If  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$ , then  $X$  has a satisfying assignment.*

*Proof.* Let us first assume that  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$ . Then, there exists an FTD-code  $S$  of  $G^\psi$  such that  $|S| = 7n + 2m$ . Then, we show that  $X$  has a satisfying assignment. Therefore, by Lemma 8.8, we only need to show that  $|\{w_1^x, w_2^x\} \cap S| = 1$  for all  $x \in X$  (note that  $|\{w_1^x, w_2^x\} \cap S| \geq 1$  in order for  $S$  to full-separate the vertices  $v_1^x, v_3^x$ ). If, on the contrary,  $|\{w_1^{x^*}, w_2^{x^*}\} \cap S| = 2$  for some  $x^* \in X$ , then, by Lemmas 8.6(a) and (b), this implies that

$$\begin{aligned} 7n + 2m = |S| &= \sum_{x \in X} |V(G^x) \cap S| + \sum_{\mathbf{c} \in B} |V(P^{\mathbf{c}}) \cap S| \\ &= \sum_{x \in X} |(V(G^x) \setminus \{w_1^x, w_2^x\}) \cap S| + \sum_{\mathbf{c} \in B} |V(P^{\mathbf{c}}) \cap S| + \sum_{x \in X} |\{w_1^x, w_2^x\} \cap S| \\ &\geq 6n + 2m + |\{w_1^{x^*}, w_2^{x^*}\} \cap S| + \sum_{x \in X \setminus \{x^*\}} |\{w_1^x, w_2^x\} \cap S| \geq 7n + 2m + 1. \end{aligned}$$

However, this is a contradiction. Therefore, we must have  $|\{w_1^x, w_2^x\} \cap S| = 1$  for all  $x \in X$ . This proves (1).

We now assume that  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$ . Therefore, there exists an FD-code  $S'$  of  $G^\psi$  such that  $|S'| = 7n + 2m - 1$ . To show that  $X$  has a satisfying assignment, again by Lemma 8.8, we only need to show that  $|\{w_1^x, w_2^x\} \cap S'| = 1$  for all  $x \in X$ . On the contrary, let us assume that  $|\{w_1^{x^*}, w_2^{x^*}\} \cap S'| = 2$  for some  $x^* \in X$ . Moreover, for all  $x \in X$ , we have  $|\{w_1^x, w_2^x\} \cap S'| \geq 1$  in order for  $S'$  to full-separate the vertices  $v_1^x, v_3^x$ . Therefore, using Lemma 8.6(c), we must have  $|V(G^x) \cap S'| \geq 7$  for all  $x \in X$ . This implies that

$$7n + 2m - 1 = |S'| = \sum_{x \in X} |V(G^x) \cap S'| + \sum_{\mathbf{c} \in B} |V(P^{\mathbf{c}}) \cap S'| \geq 7n + 2m.$$

This is again a contradiction. Therefore, we must have  $|\{w_1^x, w_2^x\} \cap S'| = 1$  for all  $x \in X$  and this proves (2).  $\square$

This brings us to the proofs of the main results in this section.

**Theorem 8.4.** *FTD-CODE is NP-complete.*

*Proof.* FTD-CODE clearly belongs to the class NP since it can be verified in polynomial-time if a given vertex subset of a graph  $G$  is an FTD-code of  $G$ . To prove NP-hardness, we show that an instance  $\psi = (X, \mathcal{C})$  with  $|X| = n$  and  $|\mathcal{C}| = m$  is a YES-instance of 3-SAT if and only if  $(G^\psi, 7n + 2m)$  is a YES-instance of FTD-CODE, where the graph  $G^\psi$  is as constructed in Reduction 8.2. In other

words, we show that  $X$  has a satisfying assignment if and only if there exists an FTD-code  $S$  of  $G^\psi$  such that  $|S| \leq 7n + 2m$ .

To prove the necessary part of the last statement, if  $X$  has a satisfying assignment, then by Lemma 8.7, we have  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$ . This implies that there exists an FTD-code  $S$  of  $G^\psi$  such that  $|S| = 7n + 2m$ . On the other hand, to prove the sufficiency, if there exists an FTD-code  $S$  of  $G^\psi$  such that  $|S| \leq 7n + 2m$ , then combining with Lemma 8.6, we have  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$ . Therefore, by Lemma 8.9(1), there exists a satisfying assignment on  $X$ . This proves the result.  $\square$

**Theorem 8.5.** *FD-CODE is NP-complete.*

*Proof.* FD-CODE clearly belongs to the class NP since it can be verified in polynomial-time if a given vertex subset of a graph  $G$  is an FD-code of  $G$ . To prove NP-hardness, we show that an instance  $\psi = (X, \mathcal{C})$  with  $|X| = n$  and  $|\mathcal{C}| = m$  is a YES-instance of 3-SAT if and only if  $(G^\psi, 7n + 2m - 1)$  is a YES-instance of FD-CODE, where the graph  $G^\psi$  is as constructed in Reduction 8.2. In other words, we show that  $X$  has a satisfying assignment if and only if there exists an FD-code  $S$  of  $G^\psi$  such that  $|S| \leq 7n + 2m - 1$ .

To prove the necessary part of the last statement, if  $X$  has a satisfying assignment, then by Lemma 8.7, we have  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$ . This implies that there exists an FD-code  $S$  of  $G^\psi$  such that  $|S| = 7n + 2m - 1$ . On the other hand, to prove the sufficiency, if there exists an FD-code  $S$  of  $G^\psi$  such that  $|S| \leq 7n + 2m - 1$ , then combining with Lemma 8.6, we have  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$ . Therefore, by Lemma 8.9(2), there exists a satisfying assignment on  $X$ . This proves the result.  $\square$

**Theorem 8.6.** *FD  $\neq$  FTD is NP-hard.*

*Proof.* We show that FD  $\neq$  FTD is NP-hard by showing that an instance  $\psi = (X, \mathcal{C})$  with  $|X| = n$  and  $|\mathcal{C}| = m$  is a YES-instance of 3-SAT if and only if  $(G^\psi, 7n + 2m)$  is a YES-instance of FD  $\neq$  FTD, where the graph  $G^\psi$  is as constructed in Reduction 8.2. In other words, we show that  $X$  has a satisfying assignment if and only if  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$  and  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$ .

The necessary condition is true since, if  $X$  has a satisfying assignment, then by Lemma 8.7, we have  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$  and  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$ . To prove the sufficiency condition, let us assume that  $\gamma^{\text{FTD}}(G^\psi) = 7n + 2m$  and  $\gamma^{\text{FD}}(G^\psi) = 7n + 2m - 1$ . Then, by Lemma 8.9,  $X$  has a satisfying assignment.  $\square$

## 8.3 Conclusion

In this chapter, we looked at the NP-hardness of the problems of OD-CODE, FD-CODE and FTD-CODE. As has been the case for the other problems of X-codes, these three problems also turn out to be NP-complete. As a further extension of these reductions, we find that there are two equally interesting problems which also turned out to be NP-hard, namely, the problems of deciding if the OD- and the OTD-numbers of a graph were the same; and that of deciding if the FD- and the FTD-numbers of a graph were the same. Our work in this chapter naturally leads to the following two further questions.

**Open Problem 8.1.** *For which graph families is X-CODE, for  $X \in \{\text{OD}, \text{FD}, \text{FTD}\}$ , polynomial-time solvable?*

**Open Problem 8.2.** *For which graph families are the problems of  $\text{OD} \neq \text{OTD}$  and  $\text{FD} \neq \text{FTD}$  polynomial-time solvable?*

## Chapter 9

# Location domination as a parameterized problem

As has been pointed out in Chapter 2, the problem LD-CODE is NP-complete, even on many restricted graph families (see Table 2.11 for an overview of the graph families where the problem is NP-complete). It is therefore natural to study the problem with respect to both natural and structural parameters to see if it has FPT algorithms with respect to these parameters. In this chapter, we address the problem of LD-CODE parameterized by both natural and structural parameters.

Structural parameterizations of the input graph in LD-CODE have been studied in [36, 37], and fine-grained complexity results regarding the number of vertices and solution size were obtained in [19, 37]. In [36], it was shown that the problem admits a linear kernel for the parameter max-leaf number, however (under standard complexity assumptions) no polynomial kernel exists for the solution size, combined with either the vertex cover number or the distance to a clique. They also provide a double-exponential kernel for the parameter distance to cluster. In the full version [37] of the paper, the same authors show that, assuming the ETH, LD-CODE admits neither a  $2^{o(k \log k)} n^{\mathcal{O}(1)}$ -time nor an  $n^{o(k)}$ -time algorithm, where  $n$  is the order of the input graph.

We now give an overview of our work in this chapter (all results in this chapter also appear in [44] (conference version accepted at the 35th International Symposium on Algorithms and Computation (ISAAC 2024)) and [45]).

**Natural parameters.** Among the natural parameters, we first look at LD-CODE parameterized by solution size and prove that the problem admits a  $2^{o(k^2)} \cdot n^{\mathcal{O}(1)}$  lower bound when parameterized by the solution size  $k$  and where  $n$  is the order of the input graph. First of all, note that LD-CODE is trivially FPT when parameterized by the solution size. Indeed, by Theorem 3.5, the graph has at most  $2^k + k$  vertices. Hence, LD-CODE admits a kernel with order  $\mathcal{O}(2^k)$ , and an FPT algorithm running in time  $2^{\mathcal{O}(k^2)}$  (See Proposition 9.1). By the next theorem, we prove that these bounds are optimal.

**Theorem 9.1.** *Unless the ETH fails, LD-CODE, parameterized by the solution size  $k$ , admits*

- *neither an algorithm running in time  $2^{o(k^2)} \cdot n^{\mathcal{O}(1)}$ ,*
- *nor a polynomial time kernelization algorithm that reduces the solution size and outputs a kernel with  $2^{o(k)}$  vertices.*

It was shown in [19] that LD-CODE parameterized by the solution size  $k$ , cannot be solved in time  $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$  on bipartite graphs, unless  $\text{W}[2] = \text{FPT}$ . Therefore, Theorem 9.1 improves upon this “no  $2^{\mathcal{O}(k)} n^{\mathcal{O}(1)}$  algorithm” bound from [19] (under  $\text{W}[2] \neq \text{FPT}$ ). Simultaneously, it also improves on a  $2^{o(k \log k)}$  ETH-based lower bound recently proved in [37].

To the best of our knowledge, LD-CODE is the first known problem to admit such an algorithmic lower bound, with a matching upper bound, when parameterized by the solution size. The only

other problems known to us, admitting similar lower bounds, are for structural parameterizations like vertex cover [2, 55, 93] or pathwidth [175, 183]. The only results known to us about ETH-based conditional lower bounds on the number of vertices in a kernel when parameterized by the solution size are for EDGE CLIQUE COVER [77] and BICLIQUE COVER [56]. Additionally, POINT LINE COVER does not admit a kernel with  $\mathcal{O}(k^{2-\epsilon})$  vertices, for any  $\epsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$  [155].

Continuing with natural parameters, we next show in the following theorem a result on the incompressibility of LD-CODE to an order  $\mathcal{O}(n^{2-\epsilon})$ , where  $n$  is the order of the input graph and  $\epsilon > 0$ .

**Theorem 9.2.** *LD-CODE does not admit a polynomial compression of order  $\mathcal{O}(n^{2-\epsilon})$  for any  $\epsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ , where  $n$  denotes the number of vertices of the input graph.*

The reduction used to prove Theorem 9.2 also yields the following results. First, it proves that LD-CODE, parameterized by the vertex cover number of the input graph and the solution size, does not admit a polynomial kernel, unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ . Our reduction is arguably a simpler argument than the one from [37] to obtain the latter result.

**Structural parameters.** We then investigate LD-CODE parameterized by structural parameters of the input graph, the first of which is treewidth. For LD-CODE parameterized by treewidth  $\text{tw}$ , we note here that by a straightforward application of Courcelle’s theorem [74], LD-CODE is FPT with the parameter treewidth (and even cliquewidth [75]). However, since the running time assured in Courcelle’s theorem [74] is very often quite large, as it is a tower of exponents whose height depends roughly on the size of the MSOL formula, it is interesting to study more efficient problem-specific algorithms when parameterized by the treewidth. There is a rich collection of problems that admit an FPT algorithm with single- or almost-single-exponential dependency with respect to treewidth, that is, of the form  $2^{\mathcal{O}(\text{tw})} \cdot n^{\mathcal{O}(1)}$  or  $2^{\mathcal{O}(\text{tw} \log(\text{tw}))} \cdot n^{\mathcal{O}(1)}$ , (see, for example, [76, Chapter 7]). There are a handful of graph problems that only admit FPT algorithms with double- or triple-exponential dependence in the treewidth [28, 106, 107, 85, 119, 160]. In these respective articles, the authors prove that this double- (respectively, triple-) dependence in the treewidth cannot be improved unless the Exponential Time Hypothesis (ETH) fails.

All the double- (or triple-) exponential lower bounds in treewidth cited above are for problems that are  $\#\text{NP}$ -complete,  $\Sigma_2^P$ -complete, or  $\Pi_2^P$ -complete. Indeed, until recently, this type of lower bounds were known only for problems that are complete for levels that are higher than NP in the polynomial hierarchy. Foucaud et al. [93] recently proved for the first time, that it is not necessary to go to higher levels of the polynomial hierarchy to achieve double-exponential lower bounds in the treewidth. The authors in [93] studied three NP-complete *metric-based graph problems*, namely METRIC DIMENSION, STRONG METRIC DIMENSION, and GEODETIC SET. They proved that these problems admit double-exponential lower bounds in  $\text{tw}$  (and, in fact in vertex cover number  $\text{vc}$  for the second problem) under the ETH. The first two of these three problems are also identification problems. Thus, in this chapter, we continue this line of research and prove similar double-exponential lower bounds for the running time of LD-CODE when parameterized by treewidth. More specifically, we prove the following.

**Theorem 9.3.** *Unless the ETH fails, LD-CODE parameterized by the treewidth  $\text{tw}$  of the input graph on  $n$  vertices does not admit an algorithm running in time  $2^{2^{o(\text{tw})}} \cdot \text{poly}(n)$ .*

Thus, Theorem 9.3 augments the short list of NP-complete problems that admit tight double-exponential lower bounds when parameterized by treewidth, and shows that the “local” non-metric-based problems (like LD-CODE) can also admit such bounds. We also remark here that the algorithmic lower bound of Theorem 9.3 holds true even with respect to treedepth, a parameter larger than treewidth. In contrast, DOMINATING SET admits an algorithm running in time  $\mathcal{O}(3^{\text{tw}} \cdot n^2)$  [159, 182].

Apart from establishing these lower bounds in Theorem 9.3, we also show that they are tight by designing treewidth-based dynamic programming schemes with matching running times. In particular, we prove the following result.

**Theorem 9.4.** *LD-CODE parameterized by the treewidth  $tw$  of the input graph on  $n$  vertices admits an algorithm running in time  $2^{2^{O(tw)}} \cdot n^{O(1)}$ .*

Our next result is an FPT algorithm for LD-CODE parameterized by the vertex cover number ( $vc$ ) of the input graph. More precisely, we prove the following theorem.

**Theorem 9.5.** *LD-CODE admits an algorithm running in time  $2^{O(vc \log vc)} \cdot n^{O(1)}$ , where  $vc$  is the vertex cover number of the input graph.*

The above result shows that unlike treewidth, for which we cannot expect a running time of the form  $2^{2^{o(tw)}}$ , if we choose  $vc$ , a parameter larger than treewidth, then we can improve the running time of our algorithm. Moreover, we show that the above result for LD-CODE also extends to the parameters ‘distance to clique’ and ‘twin-cover number’. Note that the reduction of Theorem 9.2 implies that LD-CODE cannot be solved in time  $2^{o(n)}$ , and thus, in time  $2^{o(vc)}$  (under the ETH), hence the bound of Theorem 9.5 is optimal up to a logarithmic factor.

Finally, we also provide a linear kernel for LD-CODE when parameterized by neighborhood diversity; and our last result is a linear vertex kernel when feedback edge set number ( $fes$ ) is considered as the parameter. It solves an open problem raised in [36, 37].

**Theorem 9.6.** *LD-CODE admits a kernel with  $O(fes)$  vertices and edges, where  $fes$  is the feedback edge set number of the input graph.*

Before we delve into the technical results, we recall here the statement of Lemma 4.1 from Chapter 4.

**Lemma 4.1.** *Let  $G$  be a connected graph on at least three vertices. Then  $G$  admits an optimal locating-dominating set  $S$  such that every support vertex of  $G$  is in  $S$ .*

Since the separation type considered in this chapter is clear, namely location or L-separation, we use the words “located” and “separated” interchangeably to mean that a pair of vertices of a graph are L-separated by some vertex subset of the graph.

## 9.1 LD-Code parameterized by Solution Size

In this section, we study the parameterized complexity of LD-CODE when parameterized by the solution size  $k$ . Proposition 9.1 shows that the problem admits a kernel with  $O(2^k)$  vertices, and hence a simple FPT algorithm running in time  $2^{O(k^2)}$ . We then go on to prove in Theorem 9.1 that this running time is optimal under the ETH.

**Proposition 9.1.** *LD-CODE admits a kernel with  $O(2^k)$  vertices and an algorithm running in time  $2^{O(k^2)} + O(k \log n)$ .*

*Proof.* Slater proved that for any graph  $G$  on  $n$  vertices with a locating-dominating set of order  $k$ , we have  $n \leq 2^k + k - 1$  [186]. Hence, if  $n > 2^k + k - 1$ , we can return a trivial NO instance (this check takes time  $O(k \log n)$ ). Otherwise, we have a kernel with  $O(2^k)$  vertices. In this case, we can enumerate all subsets of vertices of order  $k$ , and for each of them, check in quadratic time if it is a valid solution. Overall, this takes time  $\binom{n}{k} n^2$ ; since  $n \leq 2^k + k - 1$ , this is  $\binom{2^k + k - 1}{k} \cdot 2^{O(k)} = 2^{O(k^2)}$ .  $\square$

Next, we prove Theorem 9.1 to show that the bound in Proposition 9.1 is, in fact, optimal. To that end, we give the following reduction and prove some lemmas thereafter.

**Reduction 9.1.** To prove the theorem, we present a reduction that takes as input an instance  $\psi$ , with  $n$  variables, of 3-SAT and returns an instance  $(G, k)$  of LD-CODE such that  $|V(G)| = 2^{O(\sqrt{n})}$  and  $k = O(\sqrt{n})$ . By adding dummy variables in each set, we can assume that  $\sqrt{n}$  is an even integer. Suppose the variables are named  $x_{i,j}$  for  $i, j \in [\sqrt{n}]$ . The reduction constructs graph  $G$  as follows:

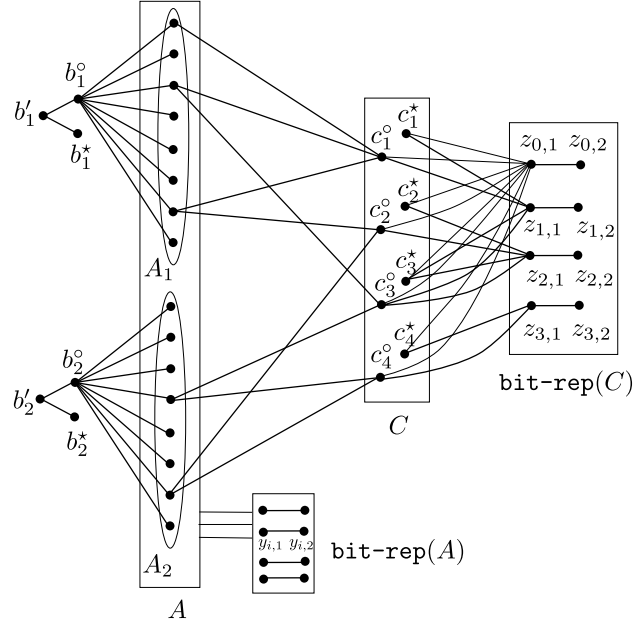


Figure 9.1: An illustrative example of the graph constructed by Reduction 9.1. Suppose an instance  $\psi$  of 3-SAT has  $n = 9$  variables and 4 clauses. We do not show the third variable bucket and explicit edges across  $A$  and  $\text{bit-rep}(A)$  for brevity.

- It partitions the variables of  $\psi$  into  $\sqrt{n}$  many *buckets*  $X_1, X_2, \dots, X_{\sqrt{n}}$  such that each bucket contains exactly  $\sqrt{n}$  many variables. Let  $X_i = \{x_{i,j} \mid j \in [\sqrt{n}]\}$  for all  $i \in [\sqrt{n}]$ .
  - For every  $X_i$ , it constructs set  $A_i$  of  $2^{\sqrt{n}}$  new vertices,  $A_i = \{a_{i,\ell} \mid \ell \in [2^{\sqrt{n}}]\}$ . Each vertex in  $A_i$  corresponds to a unique assignment of variables in  $X_i$ . Let  $A$  be the collection of all the vertices added in this step.
  - For every  $X_i$ , the reduction adds a path on three vertices  $b_i^o, b_i',$  and  $b_i^*$  with edges  $(b_i^o, b_i')$  and  $(b_i', b_i^*)$ . Suppose  $B$  is the collection of all the vertices added in this step.
  - For every  $X_i$ , it makes  $b_i^o$  adjacent with every vertex in  $A_i$ .
- For every clause  $C_j$ , the reduction adds a pair of vertices  $c_j^o, c_j^*$ . For a vertex  $a_{i,\ell} \in A_i$  for some  $i \in [\sqrt{n}]$ , and  $\ell \in [2^{\sqrt{n}}]$ , if the assignment corresponding to vertex  $a_{i,\ell}$  satisfies clause  $C_j$ , then it adds edge  $(a_{i,\ell}, c_j^o)$ .
- The reduction adds a bit-representation gadget to locate set  $A$ . Once again, informally speaking, it adds some supplementary vertices such that it is safe to assume these vertices are present in a locating-dominating set, and they locate every vertex in  $A$ . More precisely:
  - First, set  $q := \lceil \log(|A|) \rceil + 1$ . This value for  $q$  allows to uniquely represent each integer in  $[|A|]$  by its bit-representation in binary (starting from 1 and not 0).
  - For every  $i \in [q]$ , the reduction adds two vertices  $y_{i,1}$  and  $y_{i,2}$  and edge  $(y_{i,1}, y_{i,2})$ .
  - For every integer  $q' \in [|A|]$ , let  $\text{bit}(q')$  denote the binary representation of  $q'$  using  $q$  bits. Connect  $a_{i,\ell} \in A$  with  $y_{i,1}$  if the  $i^{\text{th}}$  bit in  $\text{bit}((i + (\ell - 1) \cdot \sqrt{n}))$  is 1.
  - It adds two vertices  $y_{0,1}$  and  $y_{0,2}$ , and edge  $(y_{0,1}, y_{0,2})$ . It also makes every vertex in  $A$  adjacent with  $y_{0,1}$ .

Let  $\text{bit-rep}(A)$  be the collection of the vertices  $y_{i,1}$  for all  $i \in \{0\} \cup [q]$  added in this step.
- Finally, the reduction adds a bit-representation gadget to locate set  $C$ . However, it adds the vertices in such a way that for any pair  $c_j^o, c_j^*$ , the supplementary vertices adjacent to them are identical.

- The reduction sets  $p := \lceil \log(|C|/2) \rceil + 1$  and for every  $i \in [p]$ , it adds two vertices  $z_{i,1}$  and  $z_{i,2}$  and edge  $(z_{i,1}, z_{i,2})$ .
  - For every integer  $j \in [|C|/2]$ , let  $\text{bit}(j)$  denote the binary representation of  $j$  using  $q$  bits. Connect  $c_j^\circ, c_j^\star \in C$  with  $z_{i,1}$  if the  $i^{\text{th}}$  bit in  $\text{bit}(j)$  is 1.
  - It adds two vertices  $z_{0,1}$  and  $z_{0,2}$ , and edge  $(z_{0,1}, z_{0,2})$ . It also makes every vertex in  $C$  adjacent with  $y_{0,1}$ .
- Let  $\text{bit-rep}(C)$  be the collection of the vertices  $z_{i,1}$  for all  $i \in \{0\} \cup [p]$  added in this step.

This completes the reduction. The reduction sets

$$k = |B|/3 + (\lceil \log(|A|) \rceil + 1 + 1) + \lceil \log(|C|/2) \rceil + 1 + 1 + \sqrt{n} = \mathcal{O}(\sqrt{n})$$

as  $|B| = 3\sqrt{n}$ ,  $|A| = \sqrt{n} \cdot 2^{\sqrt{n}}$ , and  $|C| = \mathcal{O}(n^3)$ , and returns  $(G, k)$  as a reduced instance.

We present a brief overview of the proof of correctness in the reverse direction. Suppose  $S$  is a locating-dominating set of graph  $G$  of order at most  $k$ . Note that  $b_i^\star$ ,  $y_{i,2}$  and  $z_{i,2}$  are pendant vertices for appropriate  $i$ . Hence, by Lemma 4.1, it is safe to consider that vertices  $b'_i$ ,  $y_{i,1}$ , and  $z_{i,1}$  are in  $S$ . This already forces  $|B|/3 + \lceil \log(|A|) \rceil + 2 + \lceil \log(|C|/2) \rceil + 2$  many vertices in  $S$ . The remaining  $\sqrt{n}$  many vertices need to locate vertices in pairs  $(b_i^\circ, b_i^\star)$  and  $(c_j^\circ, c_j^\star)$  for every  $i \in [\sqrt{n}]$  and  $j \in [|C|]$ . Note that the only vertices that are adjacent with  $b_i^\circ$  but are *not* adjacent with  $b_i^\star$  are in  $A_i$ . Also, the only vertices that are adjacent with  $c_j^\circ$  but are *not* adjacent with  $c_j^\star$  are in  $A_i$  and correspond to an assignment that satisfies  $C_j$ . Hence, any locating-dominating set should contain at least one vertex in  $A_i$  (which will locate  $b_i^\circ$  from  $b_i^\star$ ) such that the union of these vertices resolves all pairs of the form  $(c_j^\circ, c_j^\star)$ , and hence corresponds to a satisfying assignment of  $\psi$ .

**Lemma 9.1.**  *$\psi$  is a YES-instance of 3-SAT if and only if  $(G, k)$  is a YES-instance of LD-CODE.*

*Proof.* We can partition the vertices of  $G$  into the following four sets:  $B$ ,  $A$ ,  $C$ ,  $Y$ ,  $Z$ . Furthermore, we can partition  $B$  into  $B'$ ,  $B^\circ$  and  $B^\star$  as follows:  $B' = \{b'_i \mid i \in [\sqrt{n}]\}$ ,  $B^\circ = \{b_i^\circ \mid i \in [\sqrt{n}]\}$ , and  $B^\star = \{b_i^\star \mid i \in [\sqrt{n}]\}$ . Define  $Y_1$ ,  $Y_2$ ,  $Z_1$  and  $Z_2$  in the similar way. Note that  $B^\star$ ,  $Y_2$ , and  $Z_2$  together contain exactly all pendant vertices.

( $\Rightarrow$ ) Suppose  $\pi : X \mapsto \{\text{True}, \text{False}\}$  is a satisfying assignment for  $\psi$ . Using this assignment, we construct a locating-dominating set  $S$  of  $G$  of order at most  $k$ . Initialize set  $S$  by adding all the vertices in  $B' \cup Y_1 \cup Z_1$ . At this point, the cardinality of  $S$  is  $k - \sqrt{n}$ . We add the remaining  $\sqrt{n}$  vertices as follows: Partition  $X$  into  $\sqrt{n}$  parts  $X_1, \dots, X_i, \dots, X_{\sqrt{n}}$  as specified in the reduction, and define  $\pi_i$  for every  $i \in [\sqrt{n}]$  by restricting the assignment  $\pi$  to the variables in  $X_i$ . By the construction of  $G$ , there is a vertex  $a_{i,\ell}$  in  $A$  corresponding to the assignment  $\pi_i$ . Add that vertex to  $S$ . It is easy to verify that the order of  $S$  is at most  $k$ . We next argue that  $S$  is a locating-dominating set.

Consider set  $A$ . By the property of a set representation gadget, every vertex in  $A$  is adjacent with a unique set of vertices in  $\text{bit-rep}(A) \setminus \{y_{0,1}\}$ . Consider a vertex  $a_{i,\ell}$  in  $A$  such that the bit-representation of  $(i + (\ell - 1) \cdot \sqrt{n})$  contains a single 1 at the  $j^{\text{th}}$  location. Hence, both  $y_{j,2}$  and  $a_{i,\ell}$  are adjacent with the same vertex, viz  $y_{j,1}$  in  $\text{bit-rep}(R) \setminus \{y_{0,1}\}$ . However, this pair of vertices is resolved by  $y_{0,1}$  which is adjacent with  $r_i$  and *not* with  $y_{j,2}$ . Also, as the bit-representation of vertices starts from 1, there is no vertex in  $R$  which is adjacent with only  $y_{0,1}$  in  $\text{bit-rep}(R)$ . Using similar arguments and the properties of set representation gadgets, we can conclude that  $\text{bit-rep}(A) \cup \text{bit-rep}(C)$  resolves all pairs of vertices apart from pairs of the form  $(b_j^\circ, b_j^\star)$  and  $(c_j^\circ, c_j^\star)$ .

By the construction, the sets mentioned in the second row of Table 9.1 dominate the vertices mentioned in the sets in the respective first rows. Hence,  $S$  is a dominating set. We only need to argue about locating the vertices in the set that are dominated by the vertices from the set. First, consider the vertices in  $B^\circ \cup B^\star$ . Recall that every vertex of the form  $b_i^\circ$  and  $b_i^\star$  is adjacent with  $b'_i$  for every  $i \in [\sqrt{n}]$ . Hence, the only pair of vertices that needs to be located are of the form  $b_i^\circ$  and  $b_i^\star$ . However, as  $S$  contains at least one vertex from  $A_i$ , a vertex in  $S$  is adjacent with  $b_i^\circ$  and not adjacent with  $b_i^\star$ . Now, consider the vertices in  $A$  and  $Y_2$ . Note that every vertex in  $Y_2$  is adjacent



Set	$B'$	$B^\circ$	$B^*$	$A$	$Y_1$	$Y_2$	$C^\circ$	$C^*$	$Z_1$	$Z_2$
Dominated by	$B'$	$B'$	$B'$	$Y_1$	$Y_1$	$Y_1$	$Z_1$	$Z_1$	$Z_1$	$Z_1$
Located by	-	$B' + A$	$B'$	$Y_1$	-	$Y_1$	$Z_1 + A$	$Z_1$	-	$Z_1$

Table 9.1: Partition of  $V(G)$  and the corresponding set that dominates and locates the vertices in each part.

to precisely one vertex in  $Y_1$ . However, every vertex in  $A$  is adjacent with at least two vertices in  $Y_1$  (one of which is  $y_{0,1}$ ). Hence, every vertex in  $A \cup Y_2$  is located. Using similar arguments, every vertex in  $C \cup Z_2$  is also located.

The only thing that remains to argue is that every pair of vertices  $c_j^\circ$  and  $c_j^*$  is located. As  $\pi$  is a satisfying assignment, at least one of its restrictions, say  $\pi_i$ , is a satisfying assignment for clause  $C_j$ . By the construction of the graph, the vertex corresponding to  $\pi_i$  is adjacent with  $c_j^\circ$  but not adjacent with  $c_j^*$ . Also, such a vertex is present by the construction of  $S$ . Hence, there is a vertex in  $S \cap A_i$  that locates  $c_j^\circ$  from  $c_j^*$ . This concludes the proof that  $S$  is a locating-dominating set of  $G$  of order  $k$ . Hence, if  $\psi$  is a YES-instance of 3-SAT, then  $(G, k)$  is a YES-instance of LD-CODE.

( $\Leftarrow$ ) Suppose  $S$  is a locating-dominating set of  $G$  of order at most  $k$ . We construct a satisfying assignment  $\pi$  for  $\psi$ . Recall that  $B_i^*$ ,  $Y_2$ , and  $Z_2$  contain exactly all pendant vertices of  $G$ . By Lemma 4.1, it is safe to assume that every vertex in  $B'$ ,  $Y_1$ , and  $Z_1$  is present in  $S$ .

Consider the vertices in  $B^\circ$  and  $B^*$ . As mentioned before, every vertex of the form  $b_i^\circ$  and  $b_i^*$  is adjacent with  $b_i'$ , which is in  $S$ . By the construction of  $G$ , only the vertices in  $A_i$  are adjacent with  $b_i^\circ$  but not adjacent with  $b_i^*$ . Hence,  $S$  contains at least one vertex in  $A_i \cup \{b_i^\circ, b_i^*\}$ . As the number of vertices in  $S \setminus (B' \cup Y_1 \cup Y_2)$  is at most  $\sqrt{n}$ ,  $S$  contains exactly one vertex from  $A_i \cup \{b_i^\circ, b_i^*\}$  for every  $i \in [\sqrt{n}]$ . Suppose  $S$  contains a vertex from  $\{b_i^\circ, b_i^*\}$ . As the only purpose of this vertex is to locate a vertex in this set, it is safe to replace this vertex with any vertex in  $A_i$ . Hence, we can assume that  $S$  contains exactly one vertex in  $A_i$  for every  $i \in [\sqrt{n}]$ .

For every  $i \in [\sqrt{n}]$ , let  $\pi_i : X_i \mapsto \{\text{True}, \text{False}\}$  be the assignment of the variables in  $X_i$  corresponding to the vertex of  $S \cap A_i$ . We construct an assignment  $\pi : X \mapsto \{\text{True}, \text{False}\}$  such that that  $\pi$  restricted to  $X_i$  is identical to  $\pi_i$ . As  $X_i$  is a partition of variables in  $X$ , and every vertex in  $A_i$  corresponds to a valid assignment of variables in  $X_i$ , it is easy to see that  $\pi$  is a valid assignment. It remains to argue that  $\pi$  is a satisfying assignment. Consider a pair of vertices  $c_j^\circ$  and  $c_j^*$  corresponding to clause  $C_j$ . By the construction of  $G$ , both these vertices have identical neighbors in  $Z_1$ , which is contained in  $S$ . The only vertices that are adjacent with  $c_j^\circ$  and are not adjacent with  $c_j^*$  are in  $A_i$  for some  $i \in [\sqrt{n}]$  and correspond to some assignment that satisfies clause  $C_j$ . As  $S$  is a locating-dominating set of  $G$ , there is at least one vertex in  $S \cap A_i$ , that locates  $c_j^\circ$  from  $c_j^*$ . Alternately, there is at least one vertex in  $S \cap A_i$  that corresponds to an assignment that satisfies clause  $C_j$ . This implies that if  $(G, k)$  is a YES-instance of LD-CODE, then  $\phi$  is a YES-instance of 3-SAT.  $\square$

Next, we prove the main theorem of this section.

**Theorem 9.1.** *Unless the ETH fails, LD-CODE, parameterized by the solution size  $k$ , admits*

- *neither an algorithm running in time  $2^{o(k^2)} \cdot n^{O(1)}$ ,*
- *nor a polynomial time kernelization algorithm that reduces the solution size and outputs a kernel with  $2^{o(k)}$  vertices.*

*Proof.* Assume there exists an algorithm, say  $\mathcal{A}$ , that takes as input an instance  $(G, k)$  of LD-CODE and correctly concludes whether it is a YES-instance in time  $2^{o(k^2)} \cdot |V(G)|^{O(1)}$ . Consider algorithm  $\mathcal{B}$  that takes as input an instance  $\psi$  of 3-SAT, uses the reduction above to get an equivalent instance  $(G, k)$  of LD-CODE, and then uses  $\mathcal{A}$  as a subroutine. The correctness of algorithm  $\mathcal{B}$  follows from Lemma 9.1 and the correctness of algorithm  $\mathcal{A}$ . From the description of the reduction and the

fact that  $k = \sqrt{n}$ , the running time of algorithm  $\mathcal{B}$  is  $2^{\mathcal{O}(\sqrt{n})} + 2^{o(k^2)} \cdot (2^{\mathcal{O}(\sqrt{n})})^{\mathcal{O}(1)} = 2^{o(n)}$ . This, however, contradicts the ETH. Hence, LD-CODE does not admit an algorithm with running time  $2^{o(k^2)} \cdot |V(G)|^{\mathcal{O}(1)}$  unless the ETH fails.

For the second part of Theorem 9.1, assume that such a kernelization algorithm exists. Consider the following algorithm for 3-SAT. Given a 3-SAT formula on  $n$  variables, it uses the above reduction to get an equivalent instance of  $(G, k)$  such that  $|V(G)| = 2^{\mathcal{O}(\sqrt{n})}$  and  $k = \mathcal{O}(\sqrt{n})$ . Then, it uses the assumed kernelization algorithm to construct an equivalent instance  $(H, k')$  such that  $H$  has  $2^{o(k)}$  vertices and  $k' \leq k$ . Finally, it uses a brute-force algorithm, running in time  $|V(H)|^{\mathcal{O}(k')}$ , to determine whether the reduced instance, equivalently the input instance of 3-SAT, is a YES-instance. The correctness of the algorithm follows from the correctness of the respective algorithms and our assumption. The total running time of the algorithm is  $2^{\mathcal{O}(\sqrt{n})} + (|V(G)| + k)^{\mathcal{O}(1)} + |V(H)|^{\mathcal{O}(k')} = 2^{\mathcal{O}(\sqrt{n})} + (2^{\mathcal{O}(\sqrt{n})})^{\mathcal{O}(1)} + (2^{o(\sqrt{n})})^{\mathcal{O}(\sqrt{n})} = 2^{o(n)}$ . This, however, contradicts the ETH. Hence, LD-CODE does not admit a polynomial-time kernelization algorithm that reduces the solution size and returns a kernel with  $2^{o(k)}$  vertices unless the ETH fails.  $\square$

## 9.2 Incompressibility of LD-Code

In this section, we prove Theorem 9.2 for LD-CODE, that is, we prove that LD-CODE does not admit a polynomial compression of order  $\mathcal{O}(n^{2-\epsilon})$  for any  $\epsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ . We do this by transferring the incompressibility of DOMINATING SET to LD-CODE. However, it is convenient to reduce from the RED-BLUE DOMINATING SET problem, a restricted version of DOMINATING SET, which also does not admit a compression with  $\mathcal{O}(n^{2-\epsilon})$  bits unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$  [1]. RED-BLUE DOMINATING SET is formally defined as follows.

### RED-BLUE DOMINATING SET

**Input:** A bipartite graph  $G'$  with bipartition  $\langle R', B' \rangle$  and an integer  $k$ .

**Question:** Is there a set  $S' \subseteq R'$  of at most  $k'$  vertices such that  $B' \subseteq N(S')$ ?

It is safe to assume that  $G'$  does not contain an isolated vertex. Indeed, if there is an isolated vertex in  $R$ , one can delete it to obtain an equivalent instance. If  $B$  contains an isolated vertex, it is a trivial NO-instance. Moreover, it is also safe to assume that  $|R'| \leq 2^{|B'|}$  as it is safe to remove one of the two false twins in  $R'$ . Also,  $k' \leq |R'|$  as otherwise given instance is a trivial YES-instance.

**Reduction 9.2.** The reduction takes as input an instance  $(G', \langle R', B' \rangle, k')$  of RED-BLUE DOMINATING SET and constructs an instance  $(G, k)$  of LD-CODE. Suppose we have  $R' = \{r'_1, r'_2, \dots, r'_{|R'|}\}$  and  $B' = \{b'_1, b'_2, \dots, b'_{|B'|}\}$ . The reduction constructs graph  $G$  in the following steps. See Figure 9.2 for an illustration.

- It adds sets of vertices  $R$  and  $B$  to  $G$ , where  $R$  contains a vertex corresponding to each vertex in  $R'$  and  $B$  contains two vertices corresponding to each vertex in  $B'$ . Formally,  $R = \{r_i \mid i \in [|R']|\}$  and  $B = \{b_i^o, b_i^* \mid i \in [|B']|\}$ .
  - The reduction adds a *bit representation gadget* to locate set  $R$ . Informally, it adds some supplementary vertices such that it is safe to assume these vertices are present in a locating-dominating set, and they locate every vertex in  $R$ .
    - First, set  $q := \lceil \log(|R|) \rceil + 1$ . This value for  $q$  allows to uniquely represent each integer in  $[|R|]$  by its bit-representation in binary when started from 1 and not 0.
    - For every  $i \in [q]$ , the reduction adds two vertices  $y_{i,1}$  and  $y_{i,2}$  and edge  $(y_{i,1}, y_{i,2})$ .
    - For every integer  $\ell \in [|R|]$ , let  $\text{bit}(\ell)$  denote the binary representation of  $\ell$  using  $q$  bits. It connects  $r_\ell \in R$  with  $y_{i,1}$  if the  $i^{\text{th}}$  bit in  $\text{bit}(\ell)$  is 1.
    - It adds two vertices  $y_{0,1}$  and  $y_{0,2}$ , and edge  $(y_{0,1}, y_{0,2})$ , and makes every vertex in  $R$  adjacent with  $y_{0,1}$ .
- Let  $\text{bit-rep}(R)$  be the collection of the vertices  $y_{i,1}$  for all  $i \in \{0\} \cup [q]$  added in this step.

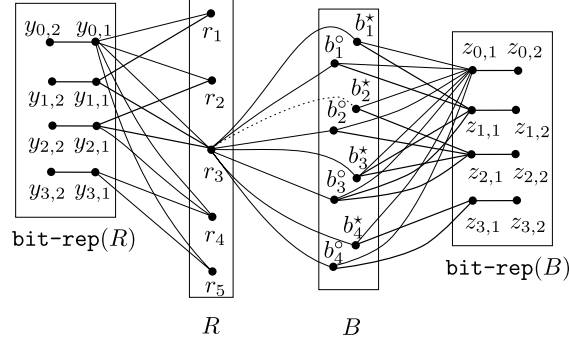


Figure 9.2: An illustrative example of the graph constructed by Reduction 9.2. Adjacency to  $y_{1,1}$ ,  $y_{2,1}$ ,  $y_{3,1}$  correspond to 1 in the bit representation from left to right (most significant to least significant) bits. In the above example, the bit representation of  $r_1$  is  $\langle 1, 0, 0 \rangle$ ,  $r_2$  is  $\langle 0, 1, 0 \rangle$ ,  $r_3$  is  $\langle 1, 1, 0 \rangle$ , etc. For brevity, we only show the edges incident on  $r_3$ . The dotted line represents a non-edge. In this example,  $r_3$  is only adjacent to  $b_2^*$  in  $G'$ . It is easy to see that  $r_3$  can only resolve pair  $(b_2^o, b_2^*)$ .

- Similarly, the reduction adds a bit representation gadget to locate set  $B$ . However, it adds vertices such that for any pair  $b_j^o, b_j^*$ , the supplementary vertices adjacent to them are identical.
  - The reduction sets  $p := \lceil \log(|B|/2) \rceil + 1$  and for every  $i \in [p]$ , it adds two vertices  $z_{i,1}$  and  $z_{i,2}$  and edge  $(z_{i,1}, z_{i,2})$ .
  - For every integer  $j \in [|B|/2]$ , let  $\text{bit}(j)$  denote the binary representation of  $j$  using  $p$  bits. Connect  $b_j^o, b_j^* \in B$  with  $z_{i,1}$  if the  $i^{\text{th}}$  bit in  $\text{bit}(j)$  is 1.
  - It add two vertices  $z_{0,1}$  and  $z_{0,2}$ , and edge  $(z_{0,1}, z_{0,2})$ . It also makes every vertex in  $B$  adjacent with  $z_{0,1}$ .
 Let  $\text{bit-rep}(B)$  be the collection of the vertices  $z_{i,1}$  for all  $i \in \{0\} \cup [p]$  added in this step.
- Finally, the reduction adds edges across  $R$  and  $B$  as follows: For every pair of vertices  $r'_i \in R'$  and  $b'_j \in B'$ ,
  - if  $r'_i$  is *not* adjacent with  $b'_j$  then it adds both edges  $(r_i, b_j^o)$  and  $(r_i, b_j^*)$  to  $E(G)$ , and
  - if  $r'_i$  is adjacent with  $b'_j$  then it adds  $(r_i, b_j^o)$  only, i.e., it does not add an edge with endpoints  $r_i$  and  $b_j^*$ .

This completes the construction of  $G$ . The reduction sets

$$k = k' + |\text{bit-rep}(R)| + |\text{bit-rep}(B)| = k' + \lceil \log(|R|) \rceil + 1 + 1 + \lceil \log(|B|/2) \rceil + 1 + 1,$$

and returns  $(G, k)$  as an instance of LD-CODE.

**Lemma 9.2.**  $(G', \langle R', B' \rangle, k')$  is a YES-instance of RED-BLUE DOMINATING SET if and only if  $(G, k)$  is a YES-instance of LD-CODE.

*Proof.* ( $\Rightarrow$ ) Suppose  $(G', \langle R', B' \rangle, k')$  is a YES-instance and let  $S' \subseteq R'$  be a solution. We prove that  $S = S' \cup \text{bit-rep}(R) \cup \text{bit-rep}(B)$  is a locating-dominating set of order at most  $k$ . The bound on order follows easily. By the property of a set representation gadget, every vertex in  $R$  is adjacent with a unique set of vertices in  $\text{bit-rep}(R) \setminus \{y_{0,1}\}$ . Consider a vertex  $r_i$  in  $R$  such that the bit-representation of  $i$  contains a single 1 at  $j^{\text{th}}$  location. Hence,  $y_{j,2}$  and  $r_i$  are adjacent with the same vertex, viz  $y_{j,1}$  in  $\text{bit-rep}(R) \setminus \{y_{0,1}\}$ . However, this pair of vertices is resolved by  $y_{0,1}$  which is adjacent with  $r_i$  and *not* with  $y_{j,2}$ . Also, as the bit-representation of vertices starts from 1, there is no vertex in  $R$  which is adjacent with only  $y_{0,1}$  in  $\text{bit-rep}(R)$ . Using similar arguments and the properties of the set representation gadget, we can conclude that  $\text{bit-rep}(R) \cup \text{bit-rep}(B)$  resolves all pairs of vertices, apart from those pairs of the form  $(b_j^o, b_j^*)$ .

By the construction of  $G$ , a vertex  $r_i \in R$  resolves a pair vertices  $b_j^\circ$  and  $b_j^*$  if and only if  $r_i'$  and  $b_j'$  are adjacent. Since  $S'$  dominates  $B$ ,  $S'$  resolves every pair of the form  $(b_j^\circ, b_j^*)$ . Hence,  $S' \cup \text{bit-rep}(R) \cup \text{bit-rep}(B)$  is a locating-dominating set of  $G$  of the desired order.

( $\Leftarrow$ ) Suppose  $S$  is a locating-dominating set of  $G$  of order at most  $k$ . By Lemma 4.1, it is safe to consider that  $S$  contains  $\text{bit-rep}(R) \cup \text{bit-rep}(B)$ , as every vertex in it is adjacent to a pendant vertex.

Next, we modify  $S$  to obtain another locating-dominating set  $S_1$  such that  $|S_1| \leq |S|$  and all vertices in  $S_1 \setminus (\text{bit-rep}(R) \cup \text{bit-rep}(B))$  are in  $R$ . Without loss of generality, suppose  $b_j^\circ \in S$ . As discussed in the previous paragraph,  $\text{bit-rep}(R) \cup \text{bit-rep}(B)$  resolves all the pair of vertices apart from the pair of the form  $(b_j^\circ, b_j^*)$ . As there is no edge between the vertices in  $B$ , if  $b_j^\circ \in S$  then it is useful only to resolve the pair  $(b_j^\circ, b_j^*)$ . As there is no isolated vertex in  $G'$ , every vertex in  $B'$  is adjacent with some vertex in  $R'$ . Hence, by the construction of  $G$ , there is a vertex  $r_i$  in  $R$  such that  $r_i$  is adjacent with  $b_j^\circ$  but not with  $b_j^*$ . Consider set  $S_1 = (S \setminus \{b_j^\circ\}) \cup r_i$ . By the arguments above,  $|S_1| \leq |S|$ ,  $S_1$  is a locating-dominating set of  $G$ , and  $|S_1 \cap B| < |S \cap B|$ . Repeating this step, we obtain set  $S_1$  with desired properties.

It is easy to verify that  $|S_1 \cap R| = |S_1 \cap R'| \leq k'$  and that for every vertex  $b_j$  in  $B'$ , there is a vertex in  $S_1 \cap R'$  which is adjacent with  $b_j$ . Hence, the order of  $S_1 \cap R$  is at most  $k$ , and it dominates every vertex in  $B$ .  $\square$

We now come to the proof of our theorem about the incompressibility of LD-CODE.

**Theorem 9.2.** *LD-CODE does not admit a polynomial compression of order  $\mathcal{O}(n^{2-\epsilon})$  for any  $\epsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ , where  $n$  denotes the number of vertices of the input graph.*

*Proof.* Suppose there exists a polynomial-time algorithm  $\mathcal{A}$  that takes as input an instance  $(G, k)$  of LD-CODE and produces an equivalent instance  $I'$ , which requires  $\mathcal{O}(n^{2-\epsilon})$  bits to encode, of some problem  $\Pi$ . Consider the compression algorithm  $\mathcal{B}$  for RED-BLUE DOMINATING SET that uses the reduction mentioned in this section and then the algorithm  $\mathcal{A}$  on the reduced instance. For an instance  $(G', \langle R', B' \rangle, k')$  of RED-BLUE DOMINATING SET, where the number of vertices in  $G'$  is  $n'$ , the reduction constructs a graph  $G$  with at most  $3n'$  vertices. Hence, for this input algorithm,  $\mathcal{B}$  constructs an equivalent instance with  $\mathcal{O}(n'^{2-\epsilon})$  bits, contradicting  $\text{NP} \subseteq \text{coNP}/\text{poly}$  [1]. Hence, LD-CODE does not admit a polynomial compression of order  $\mathcal{O}(n^{2-\epsilon})$  for any  $\epsilon > 0$ , unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ .  $\square$

**Other Consequences of Reduction 9.2.** Note that  $\text{bit-rep}(R) \cup \text{bit-rep}(B) \cup B$  is a vertex cover of  $G$  of order  $\log(|R|) + \log(|B|) + |B|$ . As  $k' \leq |B'|$  (as otherwise we are dealing with a trivially YES-instance of RED-BLUE DOMINATING SET.),  $|R'| \leq 2^{|B'|}$ , we have  $\text{vc}(G) + k \leq (\log(|R|) + \log(|B|) + |B|) + k' + \mathcal{O}(\log(|R|) + \log(|B|) + |B|) = \mathcal{O}(|B|)$ . It is known that RED-BLUE DOMINATING SET, parameterized by the number of blue vertices, does not admit a polynomial kernel unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ . See, for example, [76]. This, along with the arguments that are standard to parameter-preserving reductions, implies that LD-CODE, when parameterized jointly by the solution size and the vertex cover number of the input graph, does not admit a polynomial kernel unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ .

## 9.3 LD-Code parameterized by treewidth

In this section, we look at LD-CODE parameterized by treewidth  $\text{tw}$ . First, we present a dynamic programming algorithm that solves LD-CODE in  $2^{2^{\mathcal{O}(\text{tw})}} \cdot n^{\mathcal{O}(1)}$ . In the next section, we prove that this dependency on treewidth is optimal, up to the multiplicative constant factors, under the ETH.

### 9.3.1 Upper bound

In this subsection, we present a dynamic programming based algorithm for LD-CODE to prove Theorem 9.4 which we restate below. To that end, we refer the reader to Section 2.1.5 for the definitions of tree decomposition and treewidth of a graph.

**Theorem 9.4.** *LD-CODE, parameterized by the treewidth  $tw$  of the input graph on  $n$  vertices admits an algorithm running in time  $2^{2^{\mathcal{O}(tw)}} \cdot n^{\mathcal{O}(1)}$ .*

Since the dynamic programming is easier for tree decompositions with special properties, we introduce the classic notion of a *nice tree decomposition* [152] as follows. A rooted tree decomposition  $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$  is said to be *nice* if every node of  $T$  has at most two children and is one of the following types.

- (1) **Root node:**  $r \in V(T)$  is the root of the tree such that  $X_r = \emptyset$ .
- (2) **Leaf node:**  $t \in V(T)$  is a *leaf node* if it has no children and  $X_t = \emptyset$ .
- (3) **Introduce node:**  $t \in V(T)$  is said to be an *introduce node* if  $t$  has a unique child  $t'$  such that  $X_t = X_{t'} \uplus \{u\}$  for some  $u \in V(G)$ .
- (4) **Forget node:**  $t \in V(T)$  is said to be a *forget node* if  $t$  has a unique child  $t'$  such that  $X_{t'} = X_t \uplus \{u\}$  for some  $u \in V(G)$ .
- (5) **Join node:**  $t \in V(T)$  is said to be a *join node* if  $t$  has exactly two children  $t_1$  and  $t_2$  such that  $X_t = X_{t_1} = X_{t_2}$ .

We can omit the assumption that the input graph  $G$  is provided with a tree decomposition of width  $tw$ . We simply invoke an algorithm provided by Korhonen [153] that runs in time  $2^{\mathcal{O}(tw)}n$  and outputs a tree decomposition of width  $w \leq 2tw + 1$ . Suppose  $G$  is an undirected graph and  $\mathcal{T}$  is a tree decomposition of  $G$  having width  $w$ . Then, there exists an algorithm that runs in  $\mathcal{O}(nw^2)$ -time and outputs a nice tree decomposition of width  $w$  [152].

**Dynamic Programming:** Let  $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$  be a nice tree decomposition of  $G$  of minimum width  $tw(G)$  with root  $r$ . We are now ready to describe the details of the dynamic programming over  $\mathcal{T}$ . For every node  $t \in V(T)$ , consider the subtree  $T_t$  of  $T$  rooted at  $t$ . Let  $G_t$  denote the subgraph of  $G$  that is induced by the vertices present in the bags of  $V(G)$  corresponding to the vertices of  $T_t$ . For every node  $t \in T$ , we define a subproblem (or DP-state) using a tuple  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . Consider a partition  $(Y, W, X_t \setminus (Y \cup W))$  of  $X_t$ . The part  $Y$  denotes the vertices in the partial solution  $S$ , whereas the part  $W$  denotes the vertices of  $X_t$  dominated (but need not be located) by the solution vertices of  $G_t$  outside of  $X_t$ . To extend this partial solution, we need to keep track of vertices that are adjacent to a unique subset in  $Y$ . For example, suppose there exists a vertex  $u \in V(G_t) \setminus X_t$  such that  $N_{G_t}(u) \cap S = A$  for some subset  $A \subseteq Y$ . Then, there should not be a vertex, say  $v$ , in  $V(G) \setminus V(G_t)$  such that  $N_{G_t}(v) \cap S' = A$ , where  $S'$  is an extension of the partial solution  $S$ . Hence, we need to keep track of all such vertices by keeping track of the neighborhood of all such vertices. We define  $\mathcal{A}$ , which is a subset of  $2^Y$ , the power set of  $Y$ . The purpose of  $\mathcal{A}$  is to store all such sets that are the neighborhood of the vertices in  $V(G_t) \setminus X_t$ . Similarly, we define  $\mathcal{D}$  to store all such sets with respect to vertices that are in  $X_t$ . Finally, we define  $\mathcal{B}$  to store the pairs of vertices that are not in the partial solution  $S$  but need to be resolved by the extension of the partial solution. Formally, we define  $\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  to denote the order of a set  $S \subseteq V(G_t)$  with minimum cardinality such that

- (1)  $Y = S \cap X_t$  and  $W = N_{G_t}(S \setminus X_t) \cap X_t$ ,
- (2)  $\mathcal{A} \subseteq 2^Y$  such that  $A \in \mathcal{A}$  if and only if there exists a (unique)  $u \in V(G_t) \setminus (S \cup X_t)$  such that  $N_{G_t}(u) \cap S = A$ ,
- (3)  $\mathcal{D} \subseteq 2^Y$  such that  $A \in \mathcal{D}$  if and only if there exists  $u \in X_t \setminus S$  such that  $N_{G_t}(u) \cap S = A$ ,

- (4)  $\mathcal{B} \subseteq \binom{X_t \setminus Y}{2} \cup ((X_t \setminus Y) \times \{+\})$  such that (i) a pair of vertices  $\{x, y\} \in \mathcal{B}$  if and only if  $N_{G_t}(x) \cap S = N_{G_t}(y) \cap S$ , or (ii) the pair  $\{x, +\} \in \mathcal{B}$  if and only if there is unique  $y \in V(G_t) \setminus (S \cup X_t)$  such that  $N_{G_t}(x) \cap S = N_{G_t}(y) \cap S$ , and
- (5) there are no two vertices  $y, z \in V(G_t) \setminus (S \cup X_t)$  such that  $N_{G_t}(y) \cap S = N_{G_t}(z) \cap S$ .

In short,  $\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  denotes the optimal solution size for the DP-state or subproblem  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . If there is no set  $S \subseteq V(G_t)$  satisfying the above properties, then  $\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}] = \infty$ .

We now describe how to use the solution of the smaller subproblems to get the solution to the new subproblem. For the root node  $r \in V(T)$ ,  $X_r = \emptyset$ . If we compute the values for the following DP-states, i.e. the values  $\text{opt}[r, (\emptyset, \emptyset), (\emptyset, \emptyset), \emptyset]$ ,  $\text{opt}[r, (\emptyset, \emptyset), (\{\emptyset\}, \emptyset), \emptyset]$ ,  $\text{opt}[r, (\emptyset, \emptyset), (\emptyset, \{\emptyset\}), \emptyset]$ ,  $\text{opt}[r, (\emptyset, \emptyset), (\{\emptyset\}, \{\emptyset\}), \emptyset]$ , then we are done. Precisely, the size of an optimal solution of  $G$  is the smallest values from the above mentioned four different values. Before we illustrate our dynamic programming, we give some observations as follows.

■ **Claim 1.** Let  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  be a DP-state. If  $\{x, y\} \in \mathcal{B}$ , then either  $x, y \in W$  or  $x, y \in X_t \setminus (Y \cup W)$ .

*Proof of claim.* Let  $\{x, y\} \in \mathcal{B}$  and let  $S$  be an optimal solution to the DP-state  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . Since  $\{x, y\} \in \mathcal{B}$ , it follows that  $N_{G_t}(y) \cap S = N_{G_t}(x) \cap S$ . If  $x \in W$  then there is a vertex  $w \in S \setminus X_t$  such that  $x \in N_{G_t}(w) \cap S$ . As  $N_{G_t}(y) \cap S = N_{G_t}(x) \cap S$ , it follows that  $y$  is adjacent to  $w$  and hence  $y \in W$ .

On the other hand, if  $x \in X_t \setminus (Y \cup W)$ , then certainly  $x$  is not adjacent to any vertex in  $S \setminus X_t$ . Then, either  $x$  is adjacent to some vertices of  $Y$  or  $x$  is not adjacent to any vertex of  $Y$  either. If  $x$  is adjacent to some vertex  $w \in Y$ , then  $y$  is also adjacent to  $w \in Y$  and  $y$  is not adjacent to any vertex from  $S \setminus X_t$ . Therefore,  $y \in X_t \setminus (Y \cup W)$ . Similarly, if  $x$  is not adjacent to any vertex in  $Y$ , then  $y$  also is not adjacent to any vertex in  $Y$ . Similarly, both  $x$  and  $y$  are not adjacent to any vertex of  $S$ . Therefore,  $y \in X_t \setminus (Y \cup W)$ . This completes the proof. ■

We now describe the methodology to perform dynamic programming over nice tree decomposition.

**Leaf node.** Let  $t \in V(T)$  be a leaf node. Then,  $X_t = \emptyset$  and there are four possible subproblems of dynamic programming. These are  $\text{opt}[t, (\emptyset, \emptyset), (\emptyset, \emptyset), \emptyset]$ ,  $\text{opt}[t, (\emptyset, \emptyset), (\{\emptyset\}, \emptyset), \emptyset]$ ,  $\text{opt}[t, (\emptyset, \emptyset), (\emptyset, \{\emptyset\}), \emptyset]$ , and  $\text{opt}[t, (\emptyset, \emptyset), (\{\emptyset\}, \{\emptyset\}), \emptyset]$ . It is trivial to follow that  $\emptyset$  is the only possible solution for each of these subproblems (or DP-states). Hence, we have

$$\begin{aligned} \text{opt}[t, (\emptyset, \emptyset), (\emptyset, \emptyset), \emptyset] &= \text{opt}[t, (\emptyset, \emptyset), (\{\emptyset\}, \emptyset), \emptyset] = 0, \text{ and} \\ \text{opt}[t, (\emptyset, \emptyset), (\emptyset, \{\emptyset\}), \emptyset] &= \text{opt}[t, (\emptyset, \emptyset), (\{\emptyset\}, \{\emptyset\}), \emptyset] = 0 \end{aligned}$$

The correctness of this state is trivial.

Let  $t$  be an introduce/forget node and it has only one child  $t'$ . Then, the two DP-states  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  and  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  are *compatible* if there exists an optimal solution  $S$  for  $G_t$  satisfying the properties of  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  such that the vertex subset  $S' = S \cap V(G_{t'})$  is an optimal solution satisfying the properties of  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$ . Similarly, let  $t$  be a join node and it has two children  $t_1$  and  $t_2$ . Then, the three DP-states  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ ,  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$ , and  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$  are compatible among themselves if there are optimal solutions  $S, S_1$ , and  $S_2$  to the DP-states  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ ,  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$ , and  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$  respectively, such that  $S_1 = S \cap V(G_{t_1})$  and  $S_2 = S \cap V(G_{t_2})$ .

**Introduce Node.** Let  $t \in V(T)$  be an introduce node such that  $t'$  is its unique child, satisfying  $X_t = X_{t'} \uplus \{u\}$  for some  $u \in X_t \setminus X_{t'}$ . Let a subproblem for the node  $t'$  be  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  and a subproblem for the node  $t$  be  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . Also, assume that  $S'$  is an optimal solution to the subproblem  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$ . Observe that if  $Y \not\subseteq \{Y', Y' \cup \{u\}\}$  then the two

subproblems are clearly not compatible. Thus, we distinguish the two cases  $Y = Y' \cup \{u\}$  and  $Y = Y'$ . Depending on the case, we illustrate how to check whether the subproblem  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  is compatible with the subproblem  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ .

► **Case 5:**  $Y = Y' \cup \{u\}$ .

In this case, we must have  $W = W'$  since  $N(S \setminus X_t) \cap X_t$  remains unchanged. The following changes are highlighted. Clearly, by the subproblem definitions, if  $\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  and  $\text{opt}[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  are compatible with each other, then it must be that  $S = S' \cup \{u\}$  is the optimal solution to the subproblem definition  $\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . Consider the sets  $\mathcal{A}', \mathcal{D}' \subseteq 2^{Y'}$  and  $\mathcal{B}' \subseteq \binom{X_{t'} \setminus Y'}{2} \cup ((X_t \setminus Y) \times \{+\})$ . We illustrate how these sets  $\mathcal{A}', \mathcal{D}'$ , and  $\mathcal{B}'$  should be related to the sets  $\mathcal{A}, \mathcal{D}$  and  $\mathcal{B}$ , respectively, when  $Y = Y' \cup \{u\}$  for the corresponding two DP-states to be compatible with each other.

Let  $\hat{A} \in \mathcal{A}'$ . Then, there is a vertex  $w \in V(G_t) \setminus (S' \cup X_t)$  such that  $N(w) \cap S' = \hat{A}$ . Moreover,  $uw \notin E(G)$  since  $N_{G_t}(u) \subseteq X_t$ . Hence, the same vertex  $w \in V(G_t) \setminus (S \cup X_t)$  has the same set of neighbors  $\hat{A}$  in  $G_t$ . Therefore,  $\hat{A} \in \mathcal{A}$ . Furthermore, there cannot exist any  $v \in V(G_t) \setminus (S \cup X_t)$  such that  $v$  is adjacent to  $u$  since  $N_{G_t}(u) \subseteq X_t$ . Hence, it follows that  $\mathcal{A} = \mathcal{A}'$ .

Let  $A' \in \mathcal{D}'$  and let  $u$  be adjacent to a vertex  $v \in X_t \setminus Y$  such that  $N_{G_{t'}}(v) \cap S' = A'$ . Then,  $u \in N_{G_t}(v) \cap S$  and  $A' \cup \{u\} = N_{G_t}(v) \cap S$ . Therefore, it must be that  $A = A' \cup \{u\}$  and  $A \in \mathcal{D}$  as we have that  $N_{G_t}(v) \cap S = A$ . Otherwise, if for some  $A' \in \mathcal{D}'$ , it holds that  $A' = N_{G_{t'}}(v) \cap S'$  such that  $v$  is not adjacent to  $u$ , then  $A'$  is also in  $\mathcal{D}$ .

Let us consider  $\{x, y\} \in \mathcal{B}'$ . If  $u$  is adjacent to  $x$  but not  $y$  or  $u$  is adjacent to  $y$  but not  $x$ , then it must be that  $\{x, y\} \notin \mathcal{B}$ . Otherwise, it must be that both  $x$  and  $y$  are adjacent to  $u$  or none of them adjacent to  $u$ . Then,  $\{x, y\} \in \mathcal{B}$ . Consider  $\{x, +\} \in \mathcal{B}'$ . If  $x$  is adjacent to  $u$ , then it must be that  $\{x, +\} \notin \mathcal{B}$  implying that there is no vertex  $y \in V(G_t) \setminus (S \cup X_t)$  such that  $N_{G_t}(x) \cap S = N_{G_t}(y) \cap S$ . On the other hand, if  $x$  is not adjacent to  $u$ , then  $\{x, +\} \in \mathcal{B}$ .

Each such subproblem  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  that satisfies the above properties is compatible with the subproblem  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . Let  $\mathcal{C}$  denote the collection of all the subproblems  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  that are compatible with the subproblem  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  subject to the criteria that  $Y = Y' \cup \{u\}$ . We set

$$\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}] = 1 + \min_{\mathcal{C}} \left\{ \text{opt}[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}'] \right\}$$

◀

► **Case 6:**  $Y = Y'$ .

Then  $u \notin Y$ , and moreover since  $N_{G_t}(u) \subseteq X_t$ , by definition of  $W$ , it is not possible that  $u \in W$ . Hence  $u \notin W$  and therefore, it must be that  $u \in X_t \setminus (W \cup Y)$ . Then, it must be that  $Y = Y'$ ,  $W = W'$ , and  $S = S'$ . Observe that  $N_{G_t}(u) \subseteq X_t$  since  $X_t$  is an introduce node. Observe that the sets that are in  $\mathcal{A}'$  also remain in  $\mathcal{A}$ . Moreover, the vertices of  $G_t - X_t$  and  $G_{t'} - X_{t'}$  are the same. Hence, it must be that  $\mathcal{A} = \mathcal{A}'$ .

Consider the set  $\mathcal{D}'$ . Observe that  $N_{G_t}(u) \cap S$  is the only candidate set that can be in  $\mathcal{D}$  but not in  $\mathcal{D}'$ . But note that  $S = S'$ , implying that  $N_{G_t}(u) \cap S = N_{G_{t'}}(u) \cap S'$ . If  $N_{G_t}(u) \cap S = \emptyset$ , then there is no set that is in  $\mathcal{D} \setminus \mathcal{D}'$  and hence  $\mathcal{D} = \mathcal{D}'$ . Otherwise,  $A = N_{G_t}(u) \cap S \neq \emptyset$  and it must be that  $\mathcal{D} = \mathcal{D}' \cup \{A\}$ .

Finally, consider the sets  $\mathcal{B}'$  and  $\mathcal{B}$ . Since  $Y$  is unchanged, all pairs that are already in  $\mathcal{B}'$  must also be in  $\mathcal{B}$ . Furthermore,  $u$  has no neighbor from  $G_t - X_t$ . On the contrary, as any  $v \in W$  has a neighbor in  $S \setminus X_t$ , there cannot exist any  $v \in W$  such that  $N_{G_t}(v) \cap S = N_{G_t}(u) \cap S$ . Therefore, if there is  $v \in X_t \setminus Y$  such that  $\{u, v\} \in \mathcal{B}$ , then it must be that  $v \in X_t \setminus (Y \cup W)$ . Then, if there is  $v \in X_t \setminus (Y \cup W)$  such that  $N_{G_t}(v) \cap S = N_{G_t}(u) \cap S = N_{G_t}(u) \cap Y$ , then add the pair  $\{u, v\}$  into  $\mathcal{B}$ . We can check this condition as follows. If  $A \in \mathcal{D}$  such that there is  $v \in X_t \setminus (Y \cup W)$  satisfying  $N_{G_t}(v) \cap Y = A$ , then we put  $\{u, v\} \in \mathcal{B}$ . If  $\{v, +\} \in \mathcal{B}'$  and  $\{u, v\} \in \mathcal{B}$ , then we add  $\{u, +\}$  to the set

**B.** This completes the set of conditions under which the subproblem definition  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  is compatible with the subproblem  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$ . Let  $\mathcal{C}$  be the collection of such subproblems that are compatible with  $[t, (Y, W), (\mathcal{A}, \mathcal{D})]$ . We set

$$\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}] = \min_{\mathcal{C}} \left\{ \text{opt}[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}'] \right\}$$

◀

**Forget Node.** Let  $X_t$  be the parent of  $X_{t'}$  such that  $X_t \uplus \{u\} = X_{t'}$ . Let the subproblem definition for the node  $t'$  be  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  and the subproblem definition for the node  $t$  be  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . We illustrate how to check whether the subproblem  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  is compatible with  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  or not. Observe that the size of the solution remains the same in both states. We only need to properly illustrate how the compatibility of the DP-states can be established. If  $Y \notin \{Y', Y' \setminus \{u\}\}$ , then the two subproblems are not compatible. Thus, we distinguish two cases,  $Y = Y' \setminus \{u\}$  or  $Y = Y'$ .

► **Case 1:**  $Y = Y' \setminus \{u\}$ .

Since  $u \in S \setminus X_t$ , it must be that  $W = W' \cup (N_{G_{t'}}(u) \cap (X_t \setminus Y))$  in order for the DP-states to be compatible. The next step is to identify how the sets  $\mathcal{A}', \mathcal{B}'$  and  $\mathcal{D}'$  are related to the sets  $\mathcal{A}, \mathcal{B}$  and  $\mathcal{D}$  respectively. If  $A' \in \mathcal{A}'$  such that  $u \in A'$ , then it must be that  $A' \notin \mathcal{A}$ . In this situation, if it happens that there exists  $v \in X_t \setminus Y$  such that  $u \in N_{G_t}(v)$ , then  $v \in W$ . Otherwise, if  $A' \in \mathcal{A}'$  such that  $u \notin A'$ , then it follows that  $A' \in \mathcal{A}$ .

Similarly, if  $A' \in \mathcal{D}'$  such that  $u \in A'$ , then it must be that  $A' \notin \mathcal{D}$  for the DP-states to be compatible with each other. On the other hand, if  $A' \in \mathcal{D}'$  such that  $u \notin A'$ , it must be that  $A' \in \mathcal{D}$ . Finally, we look at pairs  $\{x, y\} \in \mathcal{B}'$  and  $\{x, +\} \in \mathcal{B}'$ . Clearly, if  $\{x, y\} \in \mathcal{B}'$ , then  $N_{G_{t'}}(x) \cap S = N_{G_{t'}}(y) \cap S$  for a candidate solution  $S$  to the DP-state  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$ . Since the candidate solution remains unchanged, it must be that  $N_{G_t}(x) \cap S = N_{G_t}(y) \cap S$ , implying that  $\{x, y\} \in \mathcal{B}$ . For a similar reason, if  $\{x, +\} \in \mathcal{B}'$ , then  $\{x, +\} \in \mathcal{B}$ . Therefore,  $\mathcal{B} = \mathcal{B}'$ .

Let  $\mathcal{C}$  denote all the subproblems  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$  that are compatible with the subproblems  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . We set

$$\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}] = \min_{\mathcal{C}} \left\{ \text{opt}[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}'] \right\}$$

◀

► **Case 2:**  $u \notin Y$  but  $u \in X_t \setminus Y$ .

Here, the candidate solution to both DP-states must remain unchanged. If there exists  $A \in \mathcal{D}'$  such that  $N_{G_{t'}}(u) \cap S = A$ , then  $A$  must be added to  $\mathcal{A}$ . On the other hand, if  $u$  is the unique vertex such that  $N_{G_{t'}}(u) \cap S = A$ , then it must satisfy that  $\mathcal{D} = \mathcal{D}' \setminus \{A\}$  and  $\mathcal{A} = \mathcal{A}' \cup \{A\}$ . Otherwise, every other set  $A \in \mathcal{A}'$  must also be in  $\mathcal{A}$ . If  $\{u, +\} \in \mathcal{B}'$ , then the DP-state  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  is not compatible with the DP-state  $[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}']$ . Therefore, it must be that  $\{u, +\} \notin \mathcal{B}'$ . Similarly, if  $\{x, u\} \in \mathcal{B}'$  for some  $x \in X_t \setminus Y$ , then add  $\{x, +\}$  into  $\mathcal{B}$ . Otherwise, for every  $\{x, y\} \in \mathcal{B}'$ , it must be that  $\{x, y\} \in \mathcal{B}$ . Similarly as before, let  $\mathcal{C}$  denote the collection of all such tuples that are compatible with  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . We set

$$\text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}] = \min_{\mathcal{C}} \left\{ \text{opt}[t', (Y', W'), (\mathcal{A}', \mathcal{D}'), \mathcal{B}'] \right\}$$

◀

**Join Node.** Let  $t$  be a join node with exactly two children  $t_1$  and  $t_2$  satisfying that  $X_t = X_{t_1} = X_{t_2}$ . Let  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  denote the subproblem definition (or DP-state). Consider the subproblems of its children as  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$  and  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$ . Since  $X_{t_1} = X_{t_2} = X_t$ ,



the first and foremost conditions for them to be compatible are  $Y = Y_1 = Y_2$ , and  $W = W_1 \cup W_2$ . Let  $S_1$  and  $S_2$  be the candidate solutions for the subproblems  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$  and  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$  respectively. We illustrate how to check whether these three subproblems are compatible with each other.

► **Case 1:**  $A \in \mathcal{A}_1 \cap \mathcal{A}_2$ .

We claim that if the subproblems  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ ,  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$  and  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$  are compatible with each other, then  $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2$  and  $\mathcal{A}_1 \cap \mathcal{A}_2 = \emptyset$ . Observe that there must exist  $u \in G_{t_1} - (S_1 \cup X_{t_1})$  such that  $N_{G_{t_1}}(u) \cap S_1 = A$ , and  $v \in G_{t_2} - (S_2 \cup X_{t_2})$  such that  $N_{G_{t_2}}(v) \cap S_2 = A$  satisfying that  $u$  and  $v$  are different vertices. Then, we say that the DP-states  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$  and  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$  are not compatible to each other. Therefore, it follows that  $\mathcal{A}_1 \cap \mathcal{A}_2 = \emptyset$ . Finally, it must be that  $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2$ . ◀

► **Case 2:** Consider the sets  $\mathcal{B}_1$  and  $\mathcal{B}_2$ .

If  $\{x, y\} \in \mathcal{B}_1$ , then due to Observation 1, either both  $x, y \in W_1$  or both  $x, y \in X_{t_1} \setminus (Y_1 \cup W_1)$ . Similarly, if  $\{x, y\} \in \mathcal{B}_2$ , then either both  $x, y \in W_2$  or both  $x, y \in X_{t_2} \setminus (Y_2 \cup W_2)$ . Furthermore, if  $\{x, y\} \in \mathcal{B}$ , then either  $x, y \in W$  or  $x, y \in X_t \setminus (Y \cup W)$ . Suppose that  $\{x, y\} \in \mathcal{B}_1 \setminus \mathcal{B}_2$ . Then, observe that the neighbors of  $x$  and  $y$  in the potential solution  $S$  in  $G_{t_2}$  (and thus in  $G_t$ ) are different. Hence,  $\{x, y\} \notin \mathcal{B}$ . By symmetry, if  $\{x, y\} \in \mathcal{B}_2 \setminus \mathcal{B}_1$ , then also  $\{x, y\} \notin \mathcal{B}$ . Hence, we consider the pairs  $\{x, y\} \in \mathcal{B}_1 \cap \mathcal{B}_2$ . We have the following subcases to justify that since  $x, y$  are not located in either of the two graphs  $G_{t_1}$  and  $G_{t_2}$ , none of these two vertices are located in  $G_t$ . The first subcase is when  $x, y \in X_{t_1} \setminus (Y_1 \cup W_1)$  and  $x, y \in X_{t_2} \setminus (Y_2 \cup W_2)$ . Then, observe that the neighbors of  $x$  and  $y$  in the sets  $S_1$  and in the sets  $S_2$  are precisely the same. Therefore,  $\{x, y\} \in \mathcal{B}$ . The next subcase is when  $x, y \in X_{t_1} \setminus (Y_1 \cup W_1)$  and  $x, y \in W_2$ . In this case here also, observe that the neighbors of  $x$  and  $y$  in the sets  $S_1$  as well as in the sets  $S_2$  are precisely the same. Hence, the neighbors of  $x$  and  $y$  in the set  $S$  are the same. Therefore,  $\{x, y\} \in \mathcal{B}$ . By similar arguments, for the other two subcases, i.e. (i)  $x, y \in X_{t_2} \setminus (Y_2 \cup W_2)$  and  $x, y \in W_1$  or (ii)  $x, y \in W_1$  and  $x, y \in W_2$ , we can argue in a similar way that  $\{x, y\} \in \mathcal{B}$ . Conversely, if  $\{x, y\} \in \mathcal{B}$ , then it can be argued in a similar way that  $\{x, y\} \in \mathcal{B}_1 \cap \mathcal{B}_2$ .

We consider next  $\{x, +\} \in \mathcal{B}_1$  such that  $x \in W_1$ . There is a vertex in  $S_1 \setminus X_{t_1}$  that is adjacent to  $x$ . Similarly, if  $\{x, +\} \in \mathcal{B}_2$  such that  $x \in W_2$ , then there is a vertex in  $S_2 \setminus X_{t_2}$  that is adjacent to  $x$ . In such a case, observe that there cannot exist any vertex  $y$  that is in  $G_t \setminus X_t$  such that  $N_{G_t}(x) = N_{G_t}(y)$ . Hence,  $\{x, +\} \notin \mathcal{B}_1$ . Therefore, if  $x \in W_1 \cap W_2$  such that  $\{x, +\} \in \mathcal{B}_1 \cap \mathcal{B}_2$ , then  $\{x, +\} \notin \mathcal{B}$ .

The next case is when  $x \in W_1 \setminus W_2$  and  $\{x, +\} \in \mathcal{B}_1$ . (The case of  $x \in W_2 \setminus W_1$  is symmetric.) The first subcase arises when  $\{x, +\} \in \mathcal{B}_2$ . Since  $x \notin W_2$ , the neighbors of  $x$  in  $G_{t_2}$  are contained in  $Y_2$ . Then, observe that there is  $y_1$  in  $G_{t_1} \setminus (S_1 \cup X_{t_1})$  whose neighbors in  $S_1$  are the same as those of  $x$ . Similarly, there is  $y_2$  in  $G_{t_2} \setminus (S_2 \cup X_{t_2})$  whose neighbors in  $S_2$  are the same as those of  $x$ . Next, we observe that in  $G_t$ , the neighbors of  $x$  in  $S$  are the same as the neighbors of  $y_1$  in  $S$  but not  $y_2$ . Therefore,  $\{x, +\} \in \mathcal{B}$ . On the other hand, if  $\{x, +\} \notin \mathcal{B}_2$ , then a similar argument applies and  $\{x, +\} \in \mathcal{B}$ .

Finally, we consider the case when  $\{x, +\} \in \mathcal{B}_1$  and  $x \notin W_1 \cup W_2$ . In such a case, if  $\{x, +\} \in \mathcal{B}_2$ , then there are two vertices  $y_1 \in G_{t_1} \setminus (S_1 \cup X_{t_1})$  and  $y_2 \in G_{t_2} \setminus (S_2 \cup X_{t_2})$  such that the neighbors in  $S_1 \cup S_2$  of  $y_1$  and  $y_2$  are the same. But, neither  $y_1$  nor  $y_2$  is in the set  $X_t$ . Therefore, if  $x \notin W_1 \cup W_2$  and  $\{x, +\} \in \mathcal{B}_1 \cap \mathcal{B}_2$ , then this makes the respective DP-states incompatible with each other, as  $y_1$  and  $y_2$  cannot be located. ◀

► **Case 3:** Consider the sets  $\mathcal{D}_1$  and  $\mathcal{D}_2$ .

If  $A \in \mathcal{D}$ , then by definition there is  $u \in X_t \setminus S$  such that  $N_{G_t}(u) \cap S = A$ . Then, it must be that  $u \in X_{t_1} \setminus S_1$  and  $u \in X_{t_2} \setminus S_2$  such that the neighborhoods of  $u$  in  $G_{t_1}$  and  $G_{t_2}$  contained in the respective solutions, are  $A_1$  and  $A_2$ , respectively. Therefore,  $A_1 = A_2 = A$ . Hence,  $A \in \mathcal{D}_1 \cap \mathcal{D}_2$ . For the converse, let  $A \in \mathcal{D}_1 \cap \mathcal{D}_2$ . Then, there exists  $u \in (X_{t_1} \setminus S_1) \cap (X_{t_2} \setminus S_2)$  such that the neighborhoods of  $u$  in  $G_{t_1}$  and  $G_{t_2}$  in the respective solution are  $A$ . Therefore,  $A \in \mathcal{D}$  concluding that  $\mathcal{D} = \mathcal{D}_1 \cap \mathcal{D}_2$  for the three DP-states to be compatible to each other.

Let  $\mathcal{C}$  denote the collection of all such triplets that are compatible to each other. Recall that  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$  denote the subproblem definition of the node  $t$  with the subproblems of its children are  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$  and  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$ . Let  $S_1$  be an optimal solution to the DP-state  $[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1]$  and  $S_2$  be an optimal solution to the DP-state  $[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2]$ . Since  $Y_1 = Y_2$  and  $W_1 \cup W_2$  are two essential requirements for these three states to be compatible triplets, note that  $S = S_1 \cup S_2$  is a solution to the DP-state  $[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}]$ . Thus, the following can be concluded:

$$\begin{aligned} & \text{opt}[t, (Y, W), (\mathcal{A}, \mathcal{D}), \mathcal{B}] \\ &= \min_{\mathcal{C}} \left\{ \text{opt}[t_1, (Y_1, W_1), (\mathcal{A}_1, \mathcal{D}_1), \mathcal{B}_1] + \text{opt}[t_2, (Y_2, W_2), (\mathcal{A}_2, \mathcal{D}_2), \mathcal{B}_2] - |Y| \right\} \end{aligned}$$

◀

**Running time.** We are yet to justify the running time to compute these DP-states. By definition,  $|X_t| \leq 2\text{tw} + 1$  for every node  $t$  of the tree decomposition. Hence, observe that there are  $2^{\mathcal{O}(\text{tw})}$  distinct choices of  $Y$  and  $2^{\mathcal{O}(\text{tw})}$  choices of  $W$ . After that, there are  $2^{2^{|Y|}}$  distinct choices of  $\mathcal{A}$  and  $\mathcal{D}$  respectively. Finally, for each of them there are  $2^{\binom{Y}{2}}$  many choices of  $\mathcal{B}$ . Since  $|Y| \leq |X_t| \leq \text{tw}$ , the total number of possible DP-states is  $2^{\mathcal{O}(\text{tw})} \cdot 2^{2^{\mathcal{O}(\text{tw})}} \cdot 2^{\mathcal{O}(\text{tw}^2)}$ . For the sake of simplicity, let  $d = 2^{\mathcal{O}(\text{tw})} \cdot 2^{2^{\mathcal{O}(\text{tw})}} \cdot 2^{\mathcal{O}(\text{tw}^2)}$ . Also note that we can check the compatibility between every pair or triplets of DP-states in at most  $\mathcal{O}(d^3)$  time. Hence, the total time taken for this procedure is  $\mathcal{O}(d^3)$ -time. Therefore, computing the values of each of the DP-states can be performed in  $2^{2^{\mathcal{O}(\text{tw})}} \cdot n^{\mathcal{O}(1)}$ -time.

## 9.3.2 Lower bound

In this section, we prove Theorem 9.3 which, combined with Theorem 9.4, shows the tightness of the double-exponential bound on the running time of LD-CODE parameterized by the treewidth  $\text{tw}$  of the input graph. We remark here that the proof of Theorem 9.3 adopts the main technique developed in [93], namely, that of devising *bit-representation gadgets* and *set representation gadgets*. We now recall Theorem 9.3.

**Theorem 9.3.** *Unless the ETH fails, LD-CODE parameterized by the treewidth  $\text{tw}$  of the input graph on  $n$  vertices does not admit an algorithm running in time  $2^{2^{\mathcal{O}(\text{tw})}} \cdot \text{poly}(n)$ .*

To prove the above theorem, we present a reduction from a variant of 3-SAT called (3,3)-SAT (see [197] and Section 2.1.4). Consider the following reduction from an instance  $\phi$  of 3-SAT with  $n$  variables and  $m$  clauses to an instance  $\psi$  of (3,3)-SAT mentioned in [197]: For every variable  $x_i$  that appears  $k > 3$  times, the reduction creates  $k$  many new variables  $x_i^1, x_i^2, \dots, x_i^k$ , replaces the  $j^{\text{th}}$  occurrence of  $x_i$  by  $x_i^j$ , and adds the series of new clauses to encode  $x_i^1 \Rightarrow x_i^2 \Rightarrow \dots \Rightarrow x_i^k \Rightarrow x_i^1$ . For an instance  $\psi$  of 3-SAT, suppose  $k_i$  denotes the number of times a variable  $x_i$  appeared in  $\phi$ . Then,  $\sum_{i \in [n]} k_i \leq 3 \cdot m$ . Hence, the reduced instance  $\psi$  of (3,3)-SAT has at most  $3m$  variables and  $4m$  clauses. Using the ETH [139] and the sparsification lemma [140], we have the following result.

**Proposition 9.2.** *(3,3)-SAT, with  $n$  variables and  $m$  clauses, does not admit an algorithm running in time  $2^{\mathcal{O}(m+n)}$ , unless the ETH fails.*

We highlight that every variable appears positively and negatively at least once. Otherwise, if a variable appears only positively (respectively, only negatively) then we can assign it **True** (respectively, **False**) and safely reduce the instance by removing the clauses containing this variable. Hence, instead of the first, second, or third appearance of the variable, we use the first positive, first negative, second positive, or second negative appearance of the variable.

**Reduction 9.3.** The reduction takes as input an instance  $\psi$  of (3,3)-SAT with  $n$  variables and outputs an instance  $(G, k)$  of LD-CODE such that  $\text{tw}(G) = \mathcal{O}(\log(n))$ . Suppose  $X = \{x_1, \dots, x_n\}$  is the collection of variables and  $C = \{C_1, \dots, C_m\}$  is the collection of clauses in  $\psi$ . Here, we consider  $\langle x_1, \dots, x_n \rangle$  and  $\langle C_1, \dots, C_m \rangle$  to be arbitrary but fixed orderings of variables and clauses in  $\psi$ . For a particular clause, the first order specifies the first, second, or third (if it exists) variable in the clause in a natural way. The second ordering specifies the first/second positive/negative appearance of variables in  $X$  in a natural way. The reduction constructs a graph  $G$  as follows.

- To construct a variable gadget for  $x_i$ , it starts with two claws  $\{\alpha_i^0, \alpha_i^1, \alpha_i^2, \alpha_i^3\}$  and  $\{\beta_i^0, \beta_i^1, \beta_i^2, \beta_i^3\}$  centered at  $\alpha_i^0$  and  $\beta_i^0$ , respectively. It then adds four vertices  $x_i^1, \neg x_i^1, x_i^2, \neg x_i^2$ , and the corresponding edges, as shown in Figure 9.3. Let  $A_i$  be the collection of these twelve vertices and  $A = \bigcup_{i=1}^n A_i$ . Define  $X_i := \{x_i^1, \neg x_i^1, x_i^2, \neg x_i^2\}$ .
- To construct a clause gadget for  $C_j$ , the reduction starts with a star graph centered at  $\gamma_j^0$  and with four leaves  $\{\gamma_j^1, \gamma_j^2, \gamma_j^3, \gamma_j^4\}$ . It then adds three vertices  $c_j^1, c_j^2, c_j^3$  and the corresponding edges shown in Figure 9.3. Let  $B_j$  be the collection of these eight vertices and define  $B = \bigcup_{j=1}^m B_j$ .
- Let  $p$  be the smallest positive integer such that  $4n \leq \binom{2p}{p}$ . Define  $\mathcal{F}_p$  as the collection of subsets of  $[2p]$  that contains exactly  $p$  integers (such a collection  $\mathcal{F}_p$  is called a *Sperner family*). Define  $\text{set-rep} : \bigcup_{i=1}^n X_i \rightarrow \mathcal{F}_p$  as an injective function by arbitrarily assigning a set in  $\mathcal{F}_p$  to a vertex  $x_i^\ell$  or  $\neg x_i^\ell$ , for every  $i \in [n]$  and  $\ell \in [2]$ . In other words, every appearance of a literal is assigned a distinct subset in  $\mathcal{F}_p$ .
- The reduction adds a *connection portal*  $V$ , which is a clique on  $2p$  vertices  $v_1, v_2, \dots, v_{2p}$ . For every vertex  $v_q$  in  $V$ , the reduction adds a pendant vertex  $u_q$  adjacent to  $v_q$ .
- For each vertex  $x_i^\ell \in X$  where  $i \in [n]$  and  $\ell \in [2]$ , the reduction adds edges  $(x_i^\ell, v_q)$  for every  $q \in \text{set-rep}(x_i^\ell)$ . Similarly, it adds edges  $(\neg x_i^\ell, v_q)$  for every  $q \in \text{set-rep}(\neg x_i^\ell)$ .
- For a clause  $C_j$ , suppose variable  $x_i$  appears positively for the  $\ell^{\text{th}}$  time as the  $r^{\text{th}}$  variable in  $C_j$ . Then, the reduction adds edges across  $B$  and  $V$  such that the vertices  $c_j^r$  and  $x_i^\ell$  have the same neighborhood in  $V$ , namely, the set  $\{v_q : q \in \text{set-rep}(x_i^\ell)\}$ . For example,  $x_i$  appears positively for the second time as the third variable in  $C_j$ . Then, the vertices  $c_j^3$  and  $x_i^2$  should have the same neighborhood in  $V$ . Similarly, it adds edges for the negative appearance of the variables.

This concludes the construction of  $G$ . The reduction sets  $k = 4n + 3m + 2p$  and returns  $(G, k)$  as the reduced instance of LD-CODE.

We now provide an overview of the proof of correctness in the reverse direction. The crux of the correctness is: Without loss of generality, all the vertices in the connection portal  $V$  are present in any locating-dominating set  $S$  of  $G$ . Consider a vertex, say  $x_i^1$ , on the ‘variable-side’ of  $S$  and a vertex, say  $c_j^1$ , on the ‘clause-side’ of  $S$ . If both of these vertices have the same neighbors in the connection portal and are not adjacent to the vertices in  $S \setminus V$ , then at least one of  $x_i^1$  or  $c_j^1$  must be included in  $S$ .

More formally, suppose  $S$  is a locating-dominating set of  $G$  of order at most  $k = 4n + 3m + 2p$ . Then, we prove that  $S$  must have exactly 4 vertices from each variable gadget and exactly 3 vertices from each clause gadget. Further,  $S$  contains either  $\{\alpha_i^i, \beta_i^i, x_i^1, x_i^2\}$  or  $\{\alpha_i^i, \beta_i^i, \neg x_i^1, \neg x_i^2\}$ , but no other combination of vertices in the variable gadget corresponding to  $x_i$ . For a clause gadget corresponding to  $C_j$ ,  $S$  contains either  $\{\gamma_j^0, c_j^2, c_j^3\}$ ,  $\{\gamma_j^0, c_j^1, c_j^3\}$ , or  $\{\gamma_j^0, c_j^1, c_j^2\}$ , but no other combination. These choices imply that  $c_j^1$ ,  $c_j^2$ , or  $c_j^3$  are not adjacent to any vertex in  $S \setminus V$ . Consider the first case and suppose  $c_j^1$  corresponds to the second positive appearance of variable  $x_i$ . By the construction, the neighborhoods of  $x_i^2$  and  $c_j^1$  in  $V$  are identical. This forces a selection of  $\{\alpha_i^i, \beta_i^i, x_i^1, x_i^2\}$  in  $S$  from the variable gadget corresponding to  $x_i$ , which corresponding to setting  $x_i$  to **True**. Hence, a locating dominating set  $S$  of order at most  $k$  implies a satisfying assignment of  $\psi$ . We formalize these intuitions in the following lemmas.

**Lemma 9.3.** *If  $\psi$  be a YES-instance of (3,3)-SAT, then  $(G, k)$  is a YES-instance of LD-CODE.*

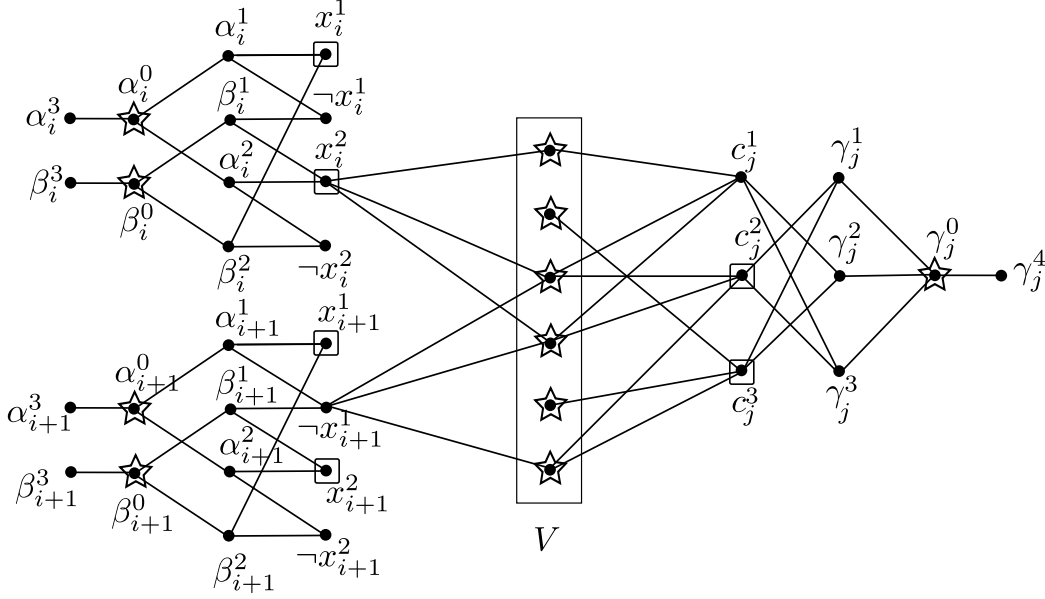


Figure 9.3: Illustration of Reduction 9.3. For the sake of clarity, we do not explicitly show the pendant vertices adjacent to vertices in  $V$ . The variable and clause gadgets are on the left-side and right-side of  $V$ , respectively. In this example, we consider a clause  $C_j = x_i \vee \neg x_{i+1} \vee x_{i+2}$ . Moreover, suppose this is the second positive appearance of  $x_i$  and the first negative appearance of  $x_{i+1}$ , and  $x_i$  corresponds to  $c_j^1$  and  $x_{i+1}$  corresponds to  $c_j^2$ . Suppose  $V$  contains 6 vertices indexed from top to bottom, and the set corresponding to these two appearances are  $\{1, 3, 4\}$  and  $\{3, 4, 6\}$  respectively. The star boundary denote the vertices that we can assume to be in any locating-dominating set, without loss of generality. The square boundary corresponds to selection of other vertices in  $S$ . In the above example, it corresponds to setting both  $x_i$  and  $x_{i+1}$  to **True**. On the clause side, the selection corresponds to selecting  $x_i$  to satisfy the clause  $C_j$ .

*Proof.* Suppose  $\pi : X \mapsto \{\text{True}, \text{False}\}$  be a satisfying assignment of  $\psi$ . We construct a vertex subset  $S$  of  $G$  from the satisfying assignment on  $\phi$  in the following manner: Initialize  $S$  by adding all the vertices in  $V$ . For variable  $x_i$ , if  $\pi(x_i) = \text{True}$ , then include  $\{\alpha_i^0, \beta_i^0, x_i^1, x_i^2\}$  in  $S$  otherwise include  $\{\alpha_i^0, \beta_i^0, \neg x_i^1, \neg x_i^2\}$  in  $S$ . For any clause  $C_j$ , if its first variable is set to **True** then include  $\{\gamma_j^0, c_j^2, c_j^3\}$  in  $S$ , if its second variable is set to **True** then include  $\{\gamma_j^0, c_j^1, c_j^3\}$  in  $S$ , otherwise include  $\{\gamma_j^0, c_j^1, c_j^2\}$  in  $S$ . If more than one variable of a clause  $C_j$  is set to **True**, we include the vertices corresponding to the smallest indexed variable set to **True**. This concludes the construction of  $S$ .

It is easy to verify that  $|S| = 4n + 3m + 2p = k$ . In the remainder of this section, we argue that  $S$  is a locating-dominating set of  $G$ . To do so, we first show that  $S$  is a dominating set of  $G$ . Notice that  $V \subseteq S$  dominates the pendant vertices  $u_q$  for all  $q \in [2p]$  and all the vertices of the form  $x_i^\ell$  for any  $i \in [n]$  and  $\ell \in [2]$ , and  $c_j^r$  for any  $j \in [m]$  and  $r \in [3]$ . Moreover, the vertices  $\alpha_i^0$ ,  $\beta_i^0$  and  $\gamma_j^0$  dominate the sets  $\{\alpha_i^1, \alpha_i^2, \alpha_i^3\}$ ,  $\{\beta_i^1, \beta_i^2, \beta_i^3\}$  and  $\{\gamma_j^1, \gamma_j^2, \gamma_j^3, \gamma_j^4\}$ , respectively. This proves that  $S$  is a dominating set of  $G$ .

We now show that  $S$  is also a locating set of  $G$ . To begin with, we notice that, for  $p \geq 2$ , all the pendant vertices  $u_q$  for  $q \in [2p]$  are located from every other vertex in  $G$  by the fact that  $|N_G(u) \cap S| = 1$ . Next, we divide the analysis of  $S$  being a locating set of  $G$  into the following three cases.

First, consider the vertices within  $A_i$ s. Each  $x_i^\ell$  or  $\neg x_i^\ell$  for any literal  $x_i \in X$  and  $\ell \in [2]$  have a distinct neighborhood in  $V$ ; hence, they are all pairwise located. For any  $i \neq i' \in [n]$  and  $\ell \neq \ell' \in [2]$ , the pair  $(\alpha_i^\ell, \beta_{i'}^{\ell'})$  is located by  $\alpha_i^0$ . Moreover, the pair  $(\alpha_i^\ell, \alpha_{i'}^{\ell'})$ , for all  $i \in [n]$ ,  $\ell \neq \ell' \in [2]$  are located by either  $\{x_i^1, x_i^2\}$  or  $\{\neg x_i^1, \neg x_i^2\}$ , one of which is a subset of  $S$ .

Second, consider the vertices within  $B_j$ s. The set of vertices in  $\{\gamma_j^0, \gamma_j^1, \gamma_j^2, \gamma_j^3, \gamma_j^4\}$  are pairwise located by three vertices in the set included in  $S$ . One of this vertex is  $\gamma_j^0$  and the other two are from  $\{c_j^1, c_j^2, c_j^3\}$ . For the one vertex in the above set which is not located by the vertices in  $S$  in the clause gadget, is located by its neighborhood in  $V$ . Finally, any two vertices of the form  $c_j^r$  and  $c_j^{r'}$ , that are not located by corresponding clause vertices in  $S$  are located from one another by the fact they are associated with two different variables or two different appearances of the same variable, and hence by construction, their neighborhood in  $V$  is different.

Third, consider the remaining vertices in clause gadget and variable gadget. Note that the only vertices in clause gadgets that we need to argue about are the vertices not adjacent with  $S$ . Suppose, let  $c_j^r$  not included in  $S$  for some  $j \in [m]$  and  $r \in [3]$ . Also, suppose  $c_j^r$  corresponds to  $\ell^{th}$  appearance of variable  $x_i$ . By the construction,  $c_j^r$  corresponds to the lowest index variable in  $C_j$  that is set to **True**. Hence, if  $x_i$  appears positively then  $x_i^\ell$  is in  $S$  whereas if it appears negatively, then  $\neg x_i^\ell$  is in  $S$ . This implies that all the remaining vertices in variable and clause gadgets are located. This proves that  $S$  is a locating set of  $G$  and, thus, the lemma.  $\square$

**Lemma 9.4.** *If  $(G, k)$  is a YES-instance of LD-CODE, then  $\psi$  be a YES-instance of (3,3)-SAT.*

*Proof.* Suppose  $S$  is a locating-dominating set of  $G$  of order  $k = 4n + 3m + 2p$ . Recall that every vertex  $v$  in the connection portal  $V$  is adjacent to a pendant vertex. Hence, by Lemma 4.1, it is safe to assume that  $S$  contains  $V$ . This implies  $|S \setminus V| = 4n + 3m$ . Using similar observation, it is safe to assume that  $\alpha_i^0, \beta_i^0$  and  $\gamma_j^0$  are in  $S$  for every  $i \in [n]$  and  $j \in [m]$ . As  $\alpha_i^3$  is adjacent with only  $\alpha_i^0$  in  $S$ , set  $S \setminus \{\alpha_i^0, \beta_i^0\}$  contains at least one vertex in the closed the neighborhood of  $\alpha_i^1$  and one from the closed neighborhood of  $\alpha_i^2$ . By the construction, these two sets are disjoint. Hence, set  $S$  contains at least four vertices from variable gadget corresponding to every variable  $x_i$  in  $X$ . Using similar arguments,  $S$  has at least three vertices from clause gadget corresponding to  $C_j$  for every clause in  $C$ . The cardinality constraints mentioned above, implies that  $S$  contains exactly four vertices from each variable gadget and exactly three vertices from each clause gadget. Using this observation, we prove the following two claims.

■ **Claim 1.** Without loss of generality, for the variable gadget corresponding to variable  $x_i$ ,  $S$  contains either  $\{\alpha_i^0, \beta_i^0, x_i^1, x_i^2\}$  or  $\{\alpha_i^0, \beta_i^0, \neg x_i^1, \neg x_i^2\}$ .

*Proof of claim.* Note that  $S$  is a locating-dominating set of  $G$  such that  $|S \cap A_i| = 4$ . Let  $X_i = \{x_i^1, x_i^2\}$  and  $\neg X_i = \{\neg x_i^1, \neg x_i^2\}$ . We prove that  $S$  contains exactly one of  $X_i$  and  $\neg X_i$ . First, we show that  $|(S \cap A_i) \setminus \{\alpha_i^0, \alpha_i^3, \beta_i^0, \beta_i^3\}| = 2$ . Define  $R_i^1 = \{\alpha_i^1, x_i^1, \neg x_i^1\}$ ,  $R_i^2 = \{\alpha_i^2, x_i^2, \neg x_i^2\}$ ,  $G_i^1 = \{\beta_i^1, \neg x_i^1, x_i^2\}$ , and  $G_i^2 = \{\beta_i^2, \neg x_i^2, x_i^1\}$ . Note that sets  $R_i^1$  and  $R_i^2$  (respectively,  $G_i^1$  and  $G_i^2$ ) are disjoint. Also,  $(R_i^1 \cap G_i^2) \cup (R_i^2 \cap G_i^1) = \{x_i^1, x_i^2\}$  and  $(R_i^1 \cap G_i^1) \cup (R_i^2 \cap G_i^2) = \{\neg x_i^1, \neg x_i^2\}$ . To distinguish the vertices in pairs  $(\alpha_i^1, \alpha_i^2)$ ,  $(\alpha_i^1, \alpha_i^3)$  and  $(\alpha_i^2, \alpha_i^3)$ , the set  $S$  must contain at least one vertex from each  $R_i^1$  and  $R_i^2$  or at least one vertex from each  $B_i^1$  and  $B_i^2$ . Similarly, to distinguish the vertices in pairs  $(\beta_i^1, \beta_i^2)$ ,  $(\beta_i^1, \beta_i^3)$  and  $(\beta_i^2, \beta_i^3)$ , the set  $S$  must contain at least one vertex from each  $R_i^1$  and  $R_i^2$  or at least one vertex from each  $B_i^1$  and  $B_i^2$ . As both of these conditions need to hold simultaneously,  $S$  contains either  $X_i$  or  $\neg X_i$ . ■

■ **Claim 2.** Without loss of generality, for the clause gadget corresponding to clause  $C_j$ ,  $S$  contains either  $\{\gamma_j^0, c_j^2, c_j^3\}$  or  $\{\gamma_j^0, c_j^1, c_j^3\}$  or  $\{\gamma_j^0, c_j^1, c_j^2\}$ .

*Proof of claim.* Note that  $S$  is a dominating set that contains 3 vertices corresponding to the clause gadget corresponding to every clause in  $C$ . Moreover,  $c_j^1, c_j^2, c_j^3$  separate the vertices in the remaining clause gadget from the set of the graph. Also, as mentioned before, without loss of generality,  $C$  contains  $\gamma_j^0$ . We define  $C_j^* = \{c_j^1, c_j^2, c_j^3\}$  and prove that  $S$  contains exactly two vertices from this set. Assume,  $S \cap C_j^* = \emptyset$ , then  $S$  needs to include all the three vertices  $\gamma_j^1, \gamma_j^2, \gamma_j^3$  to locate every pair of vertices in the clause gadget. This, however, contradicts the cardinality constraints. Assume,  $|S \cap C_j^*| = 1$  and without loss of generality, suppose  $S \cap C_j^* = \{c_j^2\}$ . If  $S \cap \{\gamma_j^1, \gamma_j^2, \gamma_j^3\} = \gamma_j^2$ , then

the pair  $(\gamma_j^1, \gamma_j^3)$  is not located by  $S$ , a contradiction. Now suppose  $S \cap \{\gamma_j^1, \gamma_j^2, \gamma_j^3\} = \gamma_j^1$ , then the pair  $(\gamma_j^2, \gamma_j^4)$  is not located by  $S$ , again a contradiction. The similar argument holds for other cases. As both cases, this leads to contradictions, we have  $|S \cap C_j^*| = 2$ . ■

Using these properties of  $S$ , we present a way to construct an assignment  $\pi$  for  $\phi$ . If  $S$  contains  $\{\alpha_i^0, \beta_i^0, x_i^1, x_i^2\}$ , then set  $\pi(x_i) = \text{True}$ , otherwise set  $\pi(x_i) = \text{False}$ . The first claim ensures that this is a valid assignment. We now prove that this assignment is also a satisfying assignment. The second claim implies that for any clause gadget corresponding to clause  $C_j$ , there is exactly one vertex *not* adjacent with any vertex in  $S \cap B_j$ . Suppose  $c_j^\ell$  be such a vertex and it is the  $\ell^{\text{th}}$  positive appearance of variable  $x_i$ . Since  $x_i^\ell$  and  $c_j^\ell$  have identical neighborhood in  $V$ , and  $S$  is a locating-dominating set in  $G$ ,  $S$  contains  $x_i^\ell$ , and hence  $\pi(x_i) = \text{True}$ . A similar reason holds when  $x_i$  appears negatively. Hence, every vertex in the clause gadget is not dominated by vertices of  $S$  in the clause gadget corresponds to the variable set to **True** that makes the clause satisfied. This implies  $\pi$  is a satisfying assignment of  $\phi$  which concludes the proof of the lemma. □

This brings us to the proof of Theorem 9.3.

**Theorem 9.3.** *Unless the ETH fails, LD-CODE parameterized by the treewidth  $tw$  of the input graph on  $n$  vertices does not admit an algorithm running in time  $2^{2^{o(tw)}} \cdot \text{poly}(n)$ .*

*Proof.* Assume there is an algorithm  $\mathcal{A}$  that, given an instance  $(G, k)$  of LD-CODE, runs in time  $2^{2^{o(tw)}} \cdot n^{\mathcal{O}(1)}$  and correctly determines whether it is YES-instance. Consider the following algorithm that takes as input an instance  $\phi$  of  $(3, 3)$ -SAT and determines whether it is a YES-instance. It first constructs an equivalent instance  $(G, k)$  of LD-CODE as mentioned in this subsection. Then, it calls algorithm  $\mathcal{A}$  as a subroutine and returns the same answer. The correctness of this algorithm follows from the correctness of algorithm  $\mathcal{A}$ , Lemma 9.3 and Lemma 9.4. Note that since each component of  $G - V$  is of constant order, the tree-width of  $G$  is  $\mathcal{O}(|V|)$ . By the asymptotic estimation of the central binomial coefficient,  $\binom{2p}{p} \sim \frac{4^p}{\sqrt{\pi \cdot p}}$  [138]. To get the upper bound of  $2p$ , we scale down the asymptotic function and have that  $4n \leq \frac{4^p}{2^p} = 2^p$ . As we choose the smallest possible value of  $p$  such that  $2^p \geq 4n$ , we can choose  $p = \log n + 3$ . Therefore,  $p = \mathcal{O}(\log(n))$ . And hence,  $|V| = \mathcal{O}(\log(n))$  which implies  $tw(G) = \mathcal{O}(\log n)$ . As the other steps, including the construction of the instance of LD-CODE, can be performed in the polynomial step, the running time of the algorithm for  $(3, 3)$ -SAT is  $2^{2^{o(\log(n))}} \cdot n^{\mathcal{O}(1)} = 2^{o(n)} \cdot n^{\mathcal{O}(1)}$ . This, however, contradicts Proposition 9.2. Hence, our assumption is wrong, and LD-CODE does not admit an algorithm running in time  $2^{2^{o(tw)}} \cdot |V(G)|^{\mathcal{O}(1)}$ , unless the ETH fails. □

## 9.4 LD-Code parameterized by vertex cover number

In this section, we prove Theorem 9.5 which we recall here.

**Theorem 9.5.** *LD-CODE admits an algorithm running in time  $2^{\mathcal{O}(vc \log vc)} \cdot n^{\mathcal{O}(1)}$ , where  $vc$  is the vertex cover number of the input graph.*

We describe an algorithm to prove Theorem 9.5. Our algorithm is based on a reduction to a dynamic programming scheme for a generalized partition refinement problem. We start with the following reduction rule which is applicable in polynomial-time.

**Reduction Rule 9.1.** *Let  $(G, k)$  be an instance of LD-CODE. If there exist three vertices  $u, v, x$  of  $G$  such that any two of  $u, v, x$  are twins, then delete  $x$  from  $G$  and decrease  $k$  by one.*

**Lemma 9.5.** *Reduction Rule 9.1 is correct and can be applied in time  $\mathcal{O}(n + m)$ .*

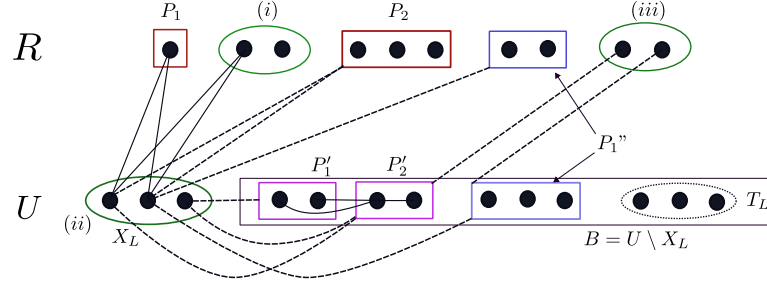


Figure 9.4: An instance  $(G, k)$  of LD-CODE. Set  $U$  is a minimum-ordered vertex cover of  $G$ . The dotted edge denotes that a vertex is adjacent with all the vertices in the set. For the sake of brevity, we do not show all the edges. The vertices in green ellipses are part of solution because of (i) being a part of false-twins, (ii) guessed intersection with  $U$ , and (iii) the requirement of solution to be a dominating set. Note that the vertices  $T_L$  are not dominated by the partial solution  $Y_L$ . Parts  $P_1, P_2, P'_1, P'_2$ , and  $P_1''$  are parts of partition of  $R \cup B$  induced by  $Y_L$ .

*Proof.* We show that  $(G, k)$  is a YES-instance of LD-CODE if and only if  $(G - \{x\}, k - 1)$  is a YES-instance of LD-CODE. Slater proved that for any set  $S$  of vertices of a graph  $G$  such that any two vertices in  $S$  are twins, any locating-dominating set contains at least  $|S| - 1$  vertices of  $S$  [186]. Hence, any locating-dominating set must contain at least two vertices in  $\{u, v, x\}$ . We can assume, without loss of generality, that any resolving set contains both  $u$  and  $x$ . Hence, any pair of vertices in  $V(G) \setminus \{u, x\}$  that is located by  $x$  is also located by  $u$ , moreover, any vertex in  $V(G) \setminus \{u, x\}$  that is dominated by  $x$  is also dominated by  $u$ . Thus, if  $S$  is a locating-dominating set of  $G$ , then  $S \setminus \{x\}$  is a locating-dominating set of  $G - \{x\}$ . This implies that if  $(G, k)$  is a YES-instance, then  $(G - \{x\}, k - 1)$  is a YES-instance. The correctness of the reverse direction follows from the fact that we can add  $x$  into a locating-dominating set of  $G - \{x\}$  to obtain one of  $G$ . Since detecting twins in a graph can be done in time  $\mathcal{O}(n + m)$  [118], the running time follows.  $\square$

Our algorithm starts by finding a minimum vertex cover, say  $U$ , of  $G$  by an algorithm in time  $1.2528^{vc} \cdot n^{\mathcal{O}(1)}$  [122]. Then,  $vc = |U|$ . Moreover, let  $R = V \setminus U$  be the corresponding independent set in  $G$ . The algorithm first applies Reduction Rule 9.1 exhaustively to reduce every twin-class to cardinality at most 2. For the simplicity of notations, we continue to call the reduced instance of LD-CODE as  $(G, k)$ .

Consider an optimal but hypothetical locating-dominating set  $L$  of  $G$ . The algorithm constructs a partial solution  $Y_L$  and, in subsequent steps, expands it to obtain  $L$ . The algorithm initializes  $Y_L$  as follows: for all pairs  $u, v$  of twins in  $G$ , it adds one of them in  $Y_L$ . Slater proved that for any set  $S$  of vertices of a graph  $G$  such that any two vertices in  $S$  are twins, any locating-dominating set contains at least  $|S| - 1$  vertices of  $S$  [186]. Hence, it is safe to assume that all the vertices in  $Y_L$  are present in any locating-dominating set. Next, the algorithm guesses the intersection of  $L$  with  $U$ . Formally, it iterates over all the subsets of  $U$ , and for each such set, say  $X_L$ , it computes a locating-dominating set of appropriate order that contains all the vertices in  $X_L$  and no vertex in  $U \setminus X_L$ . Consider a subset  $X_L$  of  $U$  and define  $B = U \setminus X_L$  and  $R' = R \setminus (N(X_L) \cup Y_L)$ . As  $L$  is also a dominating set, it is safe to assume that  $R' \subseteq L$ . The algorithm updates  $Y_L$  to include  $X_L \cup R'$ .

At this stage,  $Y_L$  dominates all the vertices in  $R$ . However, it may not dominate all the vertices in  $B$ . The remaining (to be chosen) vertices in  $L$  are part of  $R$  and are responsible for dominating the remaining vertices in  $B$  and to locate all the vertices in  $R \cup B$ . See Figure 9.4 for an illustration. As the remaining solution, i.e.,  $L \setminus Y_L$ , does not intersect  $B$ , it is safe to ignore the edges both whose endpoints are in  $B$ . The vertices in  $Y_L$  induce a partition of the vertices of the remaining graph, according to their neighborhood in  $Y_L$ . We can redefine the objective of selecting the remaining vertices in a locating-dominating set as to *refine* this partition such that each part contains exactly one vertex. Partition refinement is a classic concept in algorithms, see [118]. However in our case, the partition is not standalone, as it is induced by a solution set. To formalize this intuition, we

introduce the following notation.

**Definition 9.1** (Partition Induced by  $C$  and Refinement). For a subset  $C \subseteq V(G)$ , define an equivalence relation  $\sim_C$  on  $V \subseteq V(G) \setminus C$  as follows: for any pair  $u, v \in V$ ,  $u \sim_C v$  if and only if  $N_G(u) \cap C = N_G(v) \cap C$ . The partition of  $V$  induced by  $C$ , denoted by  $\mathcal{P}(C)$ , is the partition defined as follows:

$$\mathcal{P}(C) = \{U \mid U = \{c\} \text{ for some } c \in C \text{ or } U \text{ is an equivalence class of } \sim_C\}.$$

Moreover, for two partitions  $\mathcal{P}$  and  $\mathcal{Q}$  of  $V$ , a *refinement* of  $\mathcal{P}$  by  $\mathcal{Q}$ , denoted by  $\mathcal{P} \pitchfork \mathcal{Q}$ , is the partition defined as  $\{P \cap Q : P \in \mathcal{P}, Q \in \mathcal{Q}\}$ .

Suppose  $V = V(G)$  and  $\mathcal{P}$  and  $\mathcal{Q}$  are two partitions of  $V(G)$ , then  $\mathcal{P} \pitchfork \mathcal{Q}$  is also a partition of  $V(G)$ . We say a partition  $\mathcal{P}$  is the *identity partition* if every part of  $\mathcal{P}$  is a singleton set. With these definitions, we define the following auxiliary problem.

**ANNOTATED RED-BLUE PARTITION REFINEMENT**

**Input:** A bipartite graph  $G$  with bipartition  $\langle R, B \rangle$  of  $V(G)$ ; a partition  $\mathcal{Q}$  of  $R \cup B$ ; a collection of forced solution vertices  $C_0 \subseteq R$ ; a collection of vertices  $T_L \subseteq B$  that needs to be dominated; and an integer  $\lambda$ .

**Question:** Does there exist a set  $C$  of order at most  $\lambda$  such that  $C_0 \subseteq C \subseteq R$ ,  $C$  dominates  $T_L$ , and  $\mathcal{Q} \pitchfork \mathcal{P}(C)$  is the identity partition of  $R \cup B$ ?

Suppose there is an algorithm  $\mathcal{A}$  that solves ANNOTATED RED-BLUE PARTITION REFINEMENT in time  $f(|B|) \cdot (|R| + |B|)^{\mathcal{O}(1)}$ . Then, there is an algorithm that solves LD-CODE in time  $2^{\mathcal{O}(\text{vc})} \cdot f(\text{vc}) \cdot n^{\mathcal{O}(1)}$ . Consider the algorithm described so far in this subsection. The algorithm then calls  $\mathcal{A}$  as a subroutine with the bipartite graph obtained from  $G$  by deleting all the vertices in  $X_L$  and removing all the edges in  $B$  with  $\langle R, B \rangle$  as bipartition. It sets  $\mathcal{Q}$  as the partition of  $R \cup B$  induced by  $Y_L$ ,  $C_0 = Y_L \cap R$ ,  $T_L = B \setminus N[Y_L]$ , and  $\lambda = k - |Y_L|$ . Note that  $T_L$  is the collection of vertices that are *not* dominated by the partial solution  $Y_L$ . The correctness of the algorithm follows from the correctness of  $\mathcal{A}$  and the fact that for two sets  $C_1, C_2 \subseteq V(G)$ ,  $C_1 \cup C_2$  is a locating-dominating set of  $G$  if and only if  $\mathcal{P}(C_1) \pitchfork \mathcal{P}(C_2)$  is the identity partition of  $V(G)$ . Hence, it suffices to prove the following lemma.

**Lemma 9.6.** *There is an algorithm that solves ANNOTATED RED-BLUE PARTITION REFINEMENT in time  $2^{\mathcal{O}(|B| \log |B|)} \cdot (|R| + |B|)^{\mathcal{O}(1)}$ .*

The remainder of the section is devoted to prove Lemma 9.6. We present the algorithm that can roughly be divided into three parts. In the first part, the algorithm processes the partition  $\mathcal{Q}$  of  $R \cup B$  with the goal of reaching to a refinement of partition  $\mathcal{Q}$  such that every part is completely contained either in  $R$  or in  $B$ . Then, we introduce some terms to obtain a ‘set-cover’ type dynamic programming. We also introduce three conditions that restrict the number of dynamic programming states we need to consider to  $2^{\mathcal{O}(|B| \log(|B|))} \cdot |R|$ . Finally, we state the dynamic programming procedure, prove its correctness and argue about its running time.

**Pre-processing the Partition.** Consider the partition  $\mathcal{P}(C_0)$  of  $R \cup B$  induced by  $C_0$ , and define  $\mathcal{P}_0 = \mathcal{Q} \pitchfork \mathcal{P}(C_0)$ . We classify the parts in  $\mathcal{P}_0$  into three classes depending on whether they intersect with  $R$ ,  $B$ , or both. Let  $P_1, P_2, \dots, P_t$  be an arbitrary but fixed order on the parts of  $\mathcal{P}_0$  that are completely in  $R$ . Similarly, let  $P'_1, P'_2, \dots, P'_{t'}$  be an arbitrary but fixed order on the parts that are completely in  $B$ . Also, let  $P''_1, P''_2, \dots, P''_{t''}$  be the collection of parts that intersect  $R$  as well as  $B$ . Formally, we have  $P_j \subseteq R$  for any  $j \in [t]$ ,  $P'_{j'} \subseteq B$  for any  $j' \in [t']$ , and  $P''_{j''} \cap R \neq \emptyset \neq P''_{j''} \cap B$  for any  $j'' \in [t'']$ . See Figure 9.5 for an illustration.

Recall that  $T_L$  is the collection of vertices in  $B$  that are required to be dominated. The algorithm first expands this set so that it is precisely the collection of vertices that need to be dominated by the solution  $C$ . In other words, at present, the required condition is  $T_L \subseteq N(C)$ , whereas, after



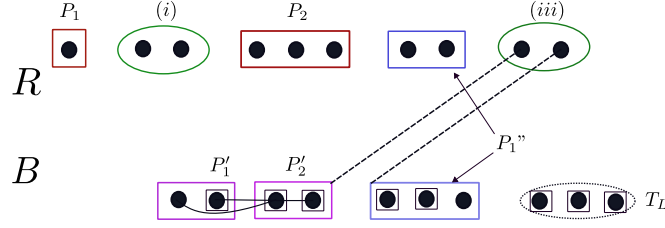


Figure 9.5: Instance of the ANNOTATED RED-BLUE PARTITION REFINEMENT problem. Vertices in green ellipse denotes set  $C_0$  whereas dotted ellipse denote vertices in  $T_L$ , and the partition  $\mathcal{Q} = \{P_1, P_2, P'_1, P'_2, P_1''\}$ . The vertices in the rectangles denote vertices in  $T_L^o$ .

the expansion, the condition is  $T_L = N(C)$ . Towards this, it first expands  $T_L$  to include  $N(C_0)$ . It then uses the property that the final partition needs to be the identity partition to add some more vertices in  $T_L$ .

Consider parts  $P'_{j'}$  or  $P''_{j''}$  of  $\mathcal{P}_0$ . Suppose that  $P'_{j'}$  (respectively,  $P''_{j''} \cap B$ ) contains two vertices that are *not* dominated by the final solution  $C$ , then these two vertices would not be separated by it and hence would remain in one part, a contradiction. Therefore, any feasible solution  $C$  needs to dominate all vertices but one in  $P'_{j'}$  (respectively in  $P'_{j'} \cap B$ ).

- The algorithm modifies  $T_L$  to include  $N(C_0)$ . Then, it iterates over all the subsets  $T_L^o$  of  $B \setminus T_L$  such that for any  $j' \in [t']$  and  $j'' \in [t'']$ ,  $|P'_{j'} \cap (T_L^o \cup T_L)| \leq 1$  and  $|(P''_{j''} \cap B) \cap (T_L^o \cup T_L)| \leq 1$ .

Consider a part  $P_j$  of  $\mathcal{P}_0$ , which is completely contained in  $R$ . As  $R$  is an independent set and the final solution  $C$  is not allowed to include any vertex in  $B$ , for any part  $P_j$ , there is at most one vertex outside  $C$ , i.e.,  $|P_j \setminus C| \leq 1$ . However, unlike the previous step, the algorithm cannot enumerate all the required subsets of  $R$  in the desired time. Nonetheless, it uses this property to safely perform the following sanity checks and modifications.

- Consider parts  $P_j$  or  $P''_{j''}$  of  $\mathcal{P}_0$ . Suppose there are two vertices in  $P_j$  (respectively  $P''_{j''} \cap R$ ) that are adjacent with some vertices in  $B \setminus T_L$ . Then, discard this guess of  $T_L^o$  and move to the next subset.
- Consider parts  $P_j$  or  $P''_{j''}$  of  $\mathcal{P}_0$ . Suppose there is a unique vertex, say  $r$ , in  $P_j$  (respectively, in  $P''_{j''} \cap R$ ) that is adjacent with some vertex in  $B \setminus T_L$ . Then, modify  $C_0$  to include all the vertices in  $P_j \setminus \{r\}$  or in  $(P''_{j''} \cup R) \setminus \{r\}$ .

Consider a part  $P''_{j''}$  of  $\mathcal{P}_0$  and suppose  $|(P''_{j''} \cap B) \setminus T_L| = 0$ . Alternately, every vertex in  $P''_{j''} \cap B$  is adjacent to some vertex in the final solution  $C$ . As  $R$  is an independent set and  $C \subseteq R$ , this domination condition already refines the partition  $P''_{j''}$ , i.e. every vertex in  $(P''_{j''} \cap B)$  needs to be adjacent with some vertex in  $C$  whereas no vertex in  $(P''_{j''} \cap R)$  can be adjacent with  $C$ . This justifies the following modification.

- For a part  $P''_{j''}$  of  $\mathcal{P}_0$  such that  $|(P''_{j''} \cap B) \setminus T_L| = 0$ , the algorithm modifies the input partition  $\mathcal{P}_0$  by removing  $P''_{j''}$  and adding two new parts  $P_j := P''_{j''} \cap R$  and  $P'_{j'} := P''_{j''} \cap B$ .

Consider a part  $P''_{j''}$  of  $\mathcal{P}_0$  and suppose  $|(P''_{j''} \cap B) \setminus T_L| = 1$ . Let  $b$  be the unique vertex in  $(P''_{j''} \cap B) \setminus T_L$ . Suppose there is a (unique) vertex, say  $r$ , in  $(P''_{j''} \cap R)$  that is not part of the final solution  $C$ . However, this implies that both  $b$  and  $r$  are in the same part, a contradiction.

- For part  $P''_{j''}$  of  $\mathcal{P}_0$  such that  $|(P''_{j''} \cap B) \setminus T_L| = 1$ , the algorithm includes all the vertices  $(P''_{j''} \cap R)$  into  $C_0$  (which, by definition, modifies  $\mathcal{P}_0$ ). Also, it removes  $P''_{j''}$  from  $\mathcal{P}_0$  and adds  $P'_{j'} := P''_{j''} \cap B$ .

These modifications ensure that any part of  $\mathcal{P}_0$  is either completely contained in  $R$  or completely contained in  $B$ .

**Setting up the Dynamic Programming.** Define  $\ell = |R \setminus C_0|$  and suppose  $\{r_1, r_2, \dots, r_\ell\}$  be the reordering of vertices in  $R \setminus C_0$  such that any part in  $P_j$  contains the consecutive elements in the order. Formally, for any  $i_1 < i_2 < i_3 \in \{1, 2, \dots, |R \setminus C_0|\}$  and  $t^\circ \in \{1, 2, \dots, t\}$ , if  $r_{i_1} \in P_{t^\circ}$  and  $r_{i_3} \in P_{t^\circ}$ , then  $r_{i_2} \in P_{t^\circ}$ . Define a function  $\pi : \{1, 2, \dots, |R \setminus C_0|\} \mapsto \{1, 2, \dots, t\}$  such that  $\pi(r_i) = t^\circ$  if and only if vertex  $r_i$  is in part  $P_{t^\circ}$ .

For the remaining part, the algorithm relies on a ‘set cover’ type dynamic programming.<sup>1</sup> For every integer  $i \in [0, \ell]$ , define  $R_i = \{r_1, r_2, \dots, r_i\}$ . Then, in a ‘set cover’-type dynamic programming, tries to find the optimum order of a subset of  $R_i$  that dominates all the vertices in  $S$ , which is a subset of  $T$ . However, two different optimum solutions in  $R_i$  will induce different partitions of  $R \cup B$ . To accommodate this, we define the notion of *valid tuples*. For an integer  $i \in [0, \ell]$  and a subset  $S \subseteq T$ , we say a tuple  $(i, \mathcal{P}, S)$  is a *valid tuple* if  $\mathcal{P}$  is a refinement of  $\mathcal{P}_0$  and it satisfies the three properties mentioned below. Note that these properties are the consequences of the fact that we require  $\mathcal{P}$  to be such that, for any (partial) solution  $C$  with vertices up to  $r_i$  in the ordering of  $R \setminus C_0$ , we can have  $\mathcal{P}(C) = \mathcal{P}$  and  $N(C) = S$ .

- (1) For any  $i \in [0, \ell]$ , the partition  $\mathcal{P}$  restricted to any  $P_j$ , where  $j \in [\pi(i) - 1]$ , is the identity partition of  $P_j$ . This is because, if on the contrary,  $\mathcal{P}$  restricted to  $P_j$  for some  $j \in [\pi(i) - 1]$  is not the identity partition, it would imply that there exists a part  $P$ , say, of  $\mathcal{P}$  with  $|P| \geq 2$  which cannot be refined by picking vertices of  $R$  in the final solution  $C$  (up to  $r_\ell$ ), as  $R$  is an independent set.
- (2) There may be at most one vertex  $r$ , say, in  $P_{\pi(i)} \cap R_i$  which is not yet picked in  $C$ . This is because if two vertices  $r', r''$  of  $P_{\pi(i)} \cap R_i$  are not picked in  $C$ , the final solution  $C$  cannot induce the identity partition, since  $\{r'\}, \{r''\} \notin \mathcal{P}(C)$ . Thus, since for any  $r' \in P_{\pi(i)} \cap R_i \setminus \{r\}$ , we have  $\{r'\} \in \mathcal{P}(C)$ , we therefore require the partition  $P$  of  $\mathcal{P}$  to be a singleton set if  $P \cap P_{\pi(i)} \cap R_i \neq \emptyset$  and  $r \notin P$ . Moreover, since  $(\{r\} \cup P_{\pi(i)} \setminus R_i) \in \mathcal{P}(C)$ , we require that for  $r \in P$ , the part  $P$  must be  $(\{r\} \cup P_{\pi(i)} \setminus R_i) \in \mathcal{P}(C)$ .
- (3) Since any (partial) solution up to  $r_i$  does not refine any of the parts  $P_{\pi(i)+1}, \dots, P_t$ , for every part  $P \in \mathcal{P}$  that does not intersect  $P_1 \cup P_2 \cup \dots \cup P_{\pi(i)}$ ,  $P$  must be of the form  $P_x$ , where  $x \in [\pi(i) + 1, t]$ .

From the above three properties of how the refinements  $\mathcal{P}$  of  $\mathcal{P}_0$  can be, we find that  $\mathcal{P}$  restricted to  $B$  can be any partition of  $B$  (which can be at most  $\mathcal{O}(2^{\mathcal{O}(|B| \log |B|)})$  many) and the sets  $P \in \mathcal{P}$  containing the vertex  $r \in P_{\pi(i)}$  can be as many as the number of vertices in  $P_{\pi(i)} \cap R_i$ , that is, at most  $|R|$  many. Let  $\mathcal{X}$  be the collection of all the valid tuples. Then, for any  $i \in [\ell]$  and any  $S \subseteq T$ , the number of partitions  $\mathcal{P}$  for which  $(i, \mathcal{P}, S) \in \mathcal{X}$  is  $\mathcal{O}(2^{\mathcal{O}(|B| \log |B|)} \cdot |R|)$ .

**Dynamic Programming.** For every valid tuple  $(i, \mathcal{P}, S) \in \mathcal{X}$ , we define

$$\begin{aligned} \text{opt}[i, \mathcal{P}, S] &= \text{the minimum cardinality of a set } C \text{ that is compatible with the} \\ &\quad \text{valid triple } (i, \mathcal{P}, S), \text{ i.e., } (i) \ C \subseteq R_i \text{ (ii) } \mathcal{P}_0 \sqcap \mathcal{P}(C) = \mathcal{P} \ \& \ \text{(iii) } N(C) = S. \end{aligned}$$

If no such  $(i, \mathcal{P}, S)$ -compatible set exists, then we let the value of  $\text{opt}[i, \mathcal{P}, S]$  to be  $\infty$ . Moreover, any  $(i, \mathcal{P}, S)$ -compatible set  $C$  of minimum cardinality is called a *minimum  $(i, \mathcal{P}, S)$ -compatible set*. The quantities  $\text{opt}[i, \mathcal{P}, S]$  for all  $(i, \mathcal{P}, S) \in \mathcal{X}$  are updated inductively. Then finally, the quantity  $\text{opt}[\ell, \mathcal{I}(R \cup B), T]$  gives us the required output of the problem. To start with, we define the quantity  $\text{opt}[0, \mathcal{P}_0, N(C_0)] = |C_0|$  and set  $\text{opt}[i, \mathcal{P}, S] = \infty$  for each triple  $(i, \mathcal{P}, S) \in \mathcal{X}$  such that  $(i, \mathcal{P}, S) \neq (0, \mathcal{P}_0, N(C_0))$ . Then, the value of each  $\text{opt}[i, \mathcal{P}, S]$  updates by the following dynamic programming formula.

$$\text{opt}[i, \mathcal{P}, S] = \min \begin{cases} \text{opt}[i-1, \mathcal{P}, S], \\ 1 + \min_{\substack{\mathcal{P}' \sqcap \mathcal{P}(r_i) = \mathcal{P}, \\ S' \cup N(r_i) = S}} \text{opt}[i-1, \mathcal{P}', S']. \end{cases} \quad (9.1)$$

<sup>1</sup>See [86, Theorem 3.10] for a simple  $\mathcal{O}(mn2^n)$  dynamic programming scheme for SET COVER on a universe of order  $n$  and  $m$  sets.

**Proof of Correctness.** We prove by induction on  $i$  that the above formula is correct. In other words, we show that, for any  $i \in [\ell]$ , if the values of  $\text{opt}[j, \mathcal{P}', S']$  are correctly calculated for all triples  $(j, \mathcal{P}', S') \in \mathcal{X}$  with  $j \in [0, i-1]$ , then for any couple  $(\mathcal{P}, S)$ , the above equality holds. Before this, we prove some technical properties.

■ **Claim 1.** Let  $(i, \mathcal{P}, S) \in \gamma^X$ . Then, the following assertions hold.

- The triple  $(i-1, \mathcal{P}, S) \in \gamma^X$  as well. Moreover, if  $C$  is a  $(i-1, \mathcal{P}, S)$ -compatible set, then  $C$  is also a  $(i, \mathcal{P}, S)$ -compatible set. In particular,  $\text{opt}[i, \mathcal{P}, S] \leq \text{opt}[i-1, \mathcal{P}, S]$ .
- Let  $(i-1, \mathcal{P}', S') \in \gamma^X$  such that  $\mathcal{P}' \cap \mathcal{P}(r_i) = \mathcal{P}$  and  $S' \cup N(r_i) = S$ . If  $C'$  is a  $(i-1, \mathcal{P}', S')$ -compatible set, then  $C = C' \cup \{r_i\}$  is a  $(i, \mathcal{P}, S)$ -compatible set. In particular,  $\text{opt}[i, \mathcal{P}, S] \leq 1 + \text{opt}[i-1, \mathcal{P}', S']$ .

*Proof of claim.* We first show that  $(i-1, \mathcal{P}, S) \in \gamma^X$ . So, let  $P \in \mathcal{P}$  be such that  $P \cap R_{i-1} \neq \emptyset$  and  $|P| \geq 2$ . It implies that  $P \cap R_i \neq \emptyset$  as well and hence,  $P \cap P_j = \emptyset$  for all  $j \in [0, \pi(i)-1]$  and  $P \cap P_{\pi(i)}$  is singleton. If  $\pi(i-1) < \pi(i)$ , then  $\pi(i-1) = \pi(i)-1$  and hence,  $P \cap P_j = \emptyset$  for all  $j \in [0, \pi(i-1)]$ . In other words,  $P \cap R_{i-1} = \emptyset$ , a contradiction to our assumption. Therefore, we must have  $\pi(i-1) = \pi(i)$  and thus, clearly,  $(i-1, \mathcal{P}, S) \in \gamma^X$ .

Now, let  $C$  is a  $(i-1, \mathcal{P}, S)$ -compatible set. Then clearly  $C$  is also a  $(i, \mathcal{P}, S)$ -compatible set. Then, the inequality follows immediately.

Now, we prove the second point. Let  $C'$  be a  $(i-1, \mathcal{P}', S')$ -compatible set. Then, we have the following.

- $C = C' \cup \{r_i\} \subset R_i$ ;
- $\mathcal{P}_0 \cap \mathcal{P}(C) = \mathcal{P}_0 \cap \mathcal{P}(C') \cap \mathcal{P}(r_i) = \mathcal{P}' \cap \mathcal{P}(r_i) = \mathcal{P}$ ; and
- $N(C) = N(C') \cup N(r_i) = S' \cup N(r_i) = S$ .

This implies that the set  $C$  is a  $(i, \mathcal{P}, S)$ -compatible set. Now, if  $\text{opt}[i-1, \mathcal{P}', S'] = \infty$ , then, clearly,  $\text{opt}[i, \mathcal{P}, S] \leq 1 + \infty = 1 + \text{opt}[i-1, \mathcal{P}, S]$ . Lastly, if  $\text{opt}[i-1, \mathcal{P}', S'] < \infty$  and  $C'$  is a minimum  $(i-1, \mathcal{P}', S')$ -compatible set, since  $C$  is  $(i, \mathcal{P}, S)$ -compatible, we have  $\text{opt}[i, \mathcal{P}, S] \leq |C| = 1 + |C'| = 1 + \text{opt}[i-1, \mathcal{P}', S']$ . ■

We now prove by induction on  $i$  that Equation (9.1) is correct. In other words, we show that, for any  $i \in [\ell]$ , if the values of  $\text{opt}[j, \mathcal{P}', S']$  are correctly calculated for all triples  $(j, \mathcal{P}', S') \in \mathcal{X}$  with  $j \in [0, i-1]$ , then for any couple  $(\mathcal{P}, S)$ , the above equality holds. Let us therefore assume that, for all triples  $(j, \mathcal{P}', S')$  with  $j \in [0, i-1]$ , the values of  $\text{opt}[j, \mathcal{P}', S']$  are correctly calculated.

To begin with, we show that if no  $(i, \mathcal{P}, S)$ -compatible set exists, then Equation (9.1) must compute  $\infty$  as the value of  $\text{opt}[i, \mathcal{P}, S]$ . In other words, we prove that the value of the RHS in Equation (9.1) is  $\infty$ . On the contrary, if  $\text{opt}[i-1, \mathcal{P}, S] < \infty$ , then there exists a  $(i-1, \mathcal{P}, S)$ -compatible set  $C$  which, by Claim 1(1), is a  $(i, \mathcal{P}, S)$ -compatible set. This contradicts our assumption in this case and therefore,  $\text{opt}[i-1, \mathcal{P}, S] = \infty$ . Similarly, if we assume that  $\text{opt}[i-1, \mathcal{P}', S'] < \infty$  for some refinement  $\mathcal{P}'$  of  $\mathcal{P}_0$  and some subset  $S'$  of  $T$  such that  $\mathcal{P}' \cap \mathcal{P}(r_i) = \mathcal{P}$  and  $S' \cup N(r_i) = S$ , then there exists a  $(i-1, \mathcal{P}', S')$ -compatible set  $C'$ . Therefore, by Claim 1(2), the set  $C = C' \cup \{r_i\}$  is a  $(i, \mathcal{P}, S)$ -compatible set. This again implies a contradiction to our assumption in this case and therefore,  $\text{opt}[i-1, \mathcal{P}', S'] = \infty$ . Hence, this proves that  $\text{RHS} = \infty$  in Equation (9.1).

In view of the above argument, therefore, we assume for the rest of the proof that  $\text{opt}[i, \mathcal{P}, S] < \infty$ . We then show that  $\text{LHS} = \text{RHS}$  in Equation (9.1). To begin with, we first show that  $\text{LHS} \leq \text{RHS}$  in Equation (9.1). This is true since, by Claim 1(1), we have  $\text{opt}[i, \mathcal{P}, S] \leq \text{opt}[i-1, \mathcal{P}, S]$  and, by Claim 1(2), for any  $(i-1, \mathcal{P}', S') \in \mathcal{X}$  such that  $\mathcal{P}' \cap \mathcal{P}(r_i) = \mathcal{P}$  and  $S' \cup N(r_i) = S$ , we have  $\text{opt}[i, \mathcal{P}, S] \leq 1 + \text{opt}[i-1, \mathcal{P}', S']$ .

We now prove the reverse, that is, the inequality  $\text{LHS} \geq \text{RHS}$  in Equation (9.1). Recall that  $\text{opt}[i, \mathcal{P}, S] < \infty$  and hence,  $\text{opt}[i, \mathcal{P}, S] = |C|$  for some minimum  $(i, \mathcal{P}, S)$ -compatible set  $C$ . We consider the following two cases.

- **Case 1** ( $r_i \notin C$ ). In this case,  $C$  is a  $(i-1, \mathcal{P}, S)$ -compatible set which implies that  $\text{RHS} \leq \text{opt}[i-1, \mathcal{P}, S] \leq |C| = \text{opt}[i, \mathcal{P}, S]$ .
- **Case 2** ( $r_i \in C$ ). Let  $C' = C \setminus \{r_i\}$ . Then,  $C'$  is a  $(i-1, \mathcal{P}', S')$ -compatible set, where  $\mathcal{P}' \cap \mathcal{P}(r_i) = \mathcal{P}$  and  $S' \cup N(r_i) = S$ . Therefore, we have  $\text{opt}[i-1, \mathcal{P}', S'] \leq |C'|$  which implies that  $\text{RHS} \leq 1 + \text{opt}[i-1, \mathcal{P}', S'] \leq 1 + |C'| = |C| = \text{opt}[i, \mathcal{P}, S]$ .

This proves that  $\text{LHS} = \text{RHS}$  in Equation (9.1) and hence the correctness of the algorithm.

**Running Time.** As mentioned before, given any  $i \in [\ell]$  and any  $S \subseteq T$ , the number of partitions  $\mathcal{P}$  for which  $(i, \mathcal{P}, S) \in \gamma^X$  is  $2^{\mathcal{O}(|B| \log |B|)} \cdot |R|$ . Moreover, the number of possible subsets  $S$  of  $T$  is at most  $2^{|T|} \in 2^{\mathcal{O}(|B|)}$ . Therefore, there are at most  $|R| \cdot 2^{\mathcal{O}(|B| \log |B|)} \cdot |R| \cdot 2^{\mathcal{O}(|B|)} \in 2^{\mathcal{O}(|B| \log |B|)} \cdot |R|^{\mathcal{O}(1)}$  many states of the dynamic programming that need to be calculated. Moreover, to calculate each such  $\text{opt}[i, \mathcal{P}, S]$ , another  $2^{\mathcal{O}(|B| \log |B|)} \cdot |R|^{\mathcal{O}(1)}$  number of  $\text{opt}[i-1, \mathcal{P}', S']$ 's are invoked according to the dynamic programming formula (9.1). As a result therefore, we obtain a total runtime of  $2^{\mathcal{O}(|B| \log |B|)} \cdot |R|^{\mathcal{O}(1)} \times 2^{\mathcal{O}(|B| \log |B|)} \cdot |R|^{\mathcal{O}(1)}$ . This completes the proof of Lemma 9.6.

## 9.5 Other structural parameters

We now consider further structural parameterizations of LD-CODE.

### 9.5.1 Extending Theorem 9.5 for twin-cover number and distance to clique

The technique designed in Section 9.4 can in fact be used to obtain algorithms for two related parameters. The *distance to clique* of a graph is the order of a smallest set of vertices that need to be removed from a graph so that the remaining vertices form a clique. It is a dense analogue of the vertex cover number (which can be seen as the “distance to independent set”). The *twin-cover number* of a graph [108] is the order of a smallest set of vertices such that the remaining vertices is a collection of disjoint cliques, each of them forming a set of mutual twins. The twin-cover number is at most the vertex cover number, since the vertices not in the vertex cover form cliques of order 1 in the remaining graph. We can obtain running in time  $2^{\mathcal{O}(\text{dc} \log \text{dc})} \cdot n^{\mathcal{O}(1)}$  and  $2^{\mathcal{O}(\text{tc} \log \text{tc})} \cdot n^{\mathcal{O}(1)}$ , where  $\text{dc}$  is the distance to clique of the input graph, and  $\text{tc}$  is the twin-cover cover number of the input graph. In both cases, the arguments are essentially the same as for  $\text{vc}$ .

For the twin-cover parameterization, let  $U$  be a set of vertices of order  $\text{tc}$  whose removal leaves a collection  $\mathcal{C}$  of cliques, each forming a twin-class. Observe that, since we have applied Reduction Rule 9.1 exhaustively, in fact every clique in  $\mathcal{C}$  has order at most 2. Moreover, recall that for any two vertices in a twin-class  $S$ , we need to select any one of them in any solution, thus those vertices are put in  $C_0$ . Note that each such selected vertex dominates its twin in  $R$ , which thus is the only vertex of  $R$  in its part of  $\mathcal{Q}$ . Apart from this, nothing changes, and we can apply Lemma 9.6 as above.

For distance to clique,  $U$  is the set of vertices of order  $\text{dc}$  whose removal leaves a clique,  $R$ . Since  $R$  forms a clique, we do no longer immediately obtain a bipartite graph, and thus we need a “split graph” version of Lemma 9.6, where  $R$  is a clique and  $B$  remains an independent set. Here, there are some slight changes during the phase of pre-processing the partition in the proof of Lemma 9.6, since any vertex in  $C$  from  $R$  dominates the whole of  $R$  and can thus potentially locate some vertex of  $R$  from some vertex of  $B$ . Nevertheless, after the pre-processing, as in Lemma 9.6, we obtain a partition where every part is completely included either in  $R$  or in  $B$ . Then, selecting a vertex of  $R$  does not distinguish any two other vertices of  $R$ , and hence the algorithm is exactly the same from that point onward.

### 9.5.2 A linear kernel for LD-Code parameterized by neighborhood diversity

We now show that LD-CODE admits a linear kernel when parameterized by the neighborhood diversity of the input graph. Recall that the *neighborhood diversity* of a graph  $G$  is the smallest integer  $\text{ndv}$  such that  $G$  can be partitioned into  $\text{ndv}$  sets of mutual twin vertices, each set being either a clique or an independent set. It was introduced in [157] and can be computed in linear time  $\mathcal{O}(n + m)$  using modular decomposition [73]. We show the following.

**Lemma 9.7.** *LD-CODE admits a kernelization algorithm that, given an instance  $(G, k)$  with neighborhood diversity  $\text{ndv}$ , it computes an equivalent instance  $(G', k')$  with  $2\text{ndv}$  vertices, and that runs in time  $\mathcal{O}(nm)$ .*

*Proof.* We repeatedly apply Reduction Rule 9.1 until there are no sets of mutual twins of order more than 2 in the graph. As each application takes time  $\mathcal{O}(n + m)$  (Lemma 9.5), overall this takes at most time  $\mathcal{O}(n(n + m)) = \mathcal{O}(nm)$ . Thus, after applying the rule, there are at most two vertices in each neighborhood equivalence class. Since there are at most  $\text{ndv}$  such classes, the resulting graph has at most  $2\text{ndv}$  vertices. By Lemma 9.5, this produces an equivalent instance  $(G', k')$ .  $\square$

**Corollary 9.1.** *LD-CODE admits an algorithm running in time  $2^{\mathcal{O}(\text{ndv})} + \mathcal{O}(nm)$ , where  $\text{ndv}$  is the neighborhood diversity of the input graph.*

*Proof.* We first compute in time  $\mathcal{O}(nm)$  the linear kernel from Theorem 9.7, that is a graph with  $n' \leq 2d$  vertices. Then, a simple brute-force algorithm running in time  $2^{\mathcal{O}(n')} = 2^{\mathcal{O}(\text{ndv})}$  completes the proof.  $\square$

## 9.6 Linear kernel for LD-Code parameterized by feedback edge set number

We next prove Theorem 9.6 which we recall as the following.

**Theorem 9.6.** *LD-CODE admits a kernel with  $\mathcal{O}(\text{fes})$  vertices and edges, where  $\text{fes}$  is the feedback edge set number of the input graph.*

Also recall that a set of edges  $X$  of a graph  $G$  is called a *feedback edge set* of  $G$  if the graph  $G - X$  is a forest. Moreover, the minimum order of a feedback edge set of  $G$  is known as the *feedback edge set number* of  $G$  and is denoted by  $\text{fes} = \text{fes}(G)$ . If a graph  $G$  has  $n$  vertices,  $m$  edges, and  $r$  components, then  $\text{fes}(G) = m - n + r$  [79].

Our proof is nonconstructive, in the sense that it shows the *existence* of a large (but constant) number of gadget types with which the kernel can be constructed, however, as there are far too many gadgets, we cannot describe them individually. We first describe graphs of given feedback edge set number.

**Proposition 9.3** ([150]). *Any graph  $G$  with feedback edge set number  $\text{fes}$  is obtained from a multigraph  $\tilde{G}$  with at most  $2\text{fes} - 2$  vertices and  $3\text{fes} - 3$  edges by first subdividing edges of  $\tilde{G}$  arbitrary number of times, and then, repeatedly attaching degree 1 vertices to the graph, and  $G$  can be computed from  $G$  in  $\mathcal{O}(n + \text{fes})$  time.*

By Proposition 9.3, we compute the multigraph  $\tilde{G}$  in  $\mathcal{O}(n + \text{fes})$  time, and we let  $\tilde{V}$  be the set of vertices of  $\tilde{G}$ . For every edge  $v_1v_2$  of  $\tilde{G}$ , we have either:

- (1)  $v_1v_2$  corresponds to an edge in the original graph  $G$ , if  $v_1v_2$  has not been subdivided when obtaining  $G$  from  $\tilde{G}$  (then  $v_1 \neq v_2$  since  $G$  is loop-free), or

- (2)  $v_1v_2$  corresponds to a component  $C$  of the original graph  $G$  with  $\{v_1, v_2\}$  removed. Moreover,  $C$  induces a tree, with two (not necessarily distinct) vertices  $c_1, c_2$  in  $C$ , where  $c_1$  is the only vertex of  $C$  adjacent to  $v_1$  and  $c_2$  is the only vertex of  $C$  adjacent to  $v_2$  in  $G$ . (Possibly,  $v_1 = v_2$  and the edge  $v_1v_2$  is a loop of  $\tilde{G}$ .)

To avoid dealing with the case of loops, for every component  $C$  of  $G \setminus \{v_1\}$  with a loop at  $v_1$  in  $\tilde{G}$  and with  $c_1, c_2$  defined as above, we select an arbitrary vertex  $x$  of the unique path between  $c_1$  and  $c_2$  in  $G[C]$  and add  $x$  to  $\tilde{V}$ . As there are at most  $\text{fes}$  loops in  $\tilde{G}$ , we can assume that  $\tilde{G}$  is a loopless multigraph with at most  $3\text{fes} - 2$  vertices and  $4\text{fes} - 3$  edges.

**Observation 9.1.** *We may assume that the multigraph  $\tilde{G}$  computed in Proposition 9.3 has at most  $3\text{fes} - 2$  vertices and  $4\text{fes} - 3$  edges, and no loops.*

The kernelization has two steps. First, we need to handle *hanging trees*, that is, parts of the graph that induce trees and are connected to the rest of the graph via a single edge. Those correspond to the iterated addition of degree 1 vertices to the graph from  $\tilde{G}$  described in Proposition 9.3. Step 2 will deal with the subgraphs that correspond to the edges of  $\tilde{G}$ .

**Step 1 – Handling the Hanging Trees.** It is known that there is a linear-time dynamic programming algorithm to solve LD-CODE on trees [185] (see also [10] for a more general algorithm for block graphs, that also solves several types of problems related to LD-CODE). To this end, one can in fact define the following five types of (partial) solutions. For a tree  $T$  rooted at a vertex  $v$ , we define five types for a subset  $L$  of  $V(T)$  as follows:

- Type *A*:  $L$  dominates all vertices of  $V(T) \setminus \{v\}$  and any two vertices of  $V(T) \setminus (L \cup \{v\})$  are located.
- Type *B*:  $L$  is a dominating set of  $T$  and any two vertices of  $V(T) \setminus (L \cup \{v\})$  are located;
- Type *C*:  $L$  is a locating-dominating set of  $T$ ;
- Type *D*:  $L$  is a locating-dominating set of  $T$  with  $v \in L$ ;
- Type *E*:  $L$  is a locating-dominating set of  $T$  such that  $v \in L$  and there is no vertex  $w$  of  $T$  with  $N(w) \cap L = \{v\}$ ;

Next, for convenience, we consider the lexicographic ordering  $A < B < C < D < E$ . Note that these five types of solution are increasingly constrained (so, if  $X < X'$ , if  $L$  is a solution of type  $X'$ , it is also one of type  $X$ ). For a tree  $T$  rooted at  $v$  and  $X \in \{A, B, C, D, E\}$ , we denote by  $\text{opt}_X(T, v)$  the minimum order of a set  $L \subseteq V(T)$  of type  $X$ . Note that  $\text{opt}_C(T, v)$  is equal to the location-domination number of  $T$ .

The following proposition can be proved by using the same ideas as in the dynamic programming algorithms from [10, 185].

**Proposition 9.4.** *There is a linear-time dynamic programming algorithm that, for any tree  $T$  and root  $v$ , conjointly computes  $\text{opt}_X(T, v)$  for all  $X \in \{A, B, C, D, E\}$ .*

**Lemma 9.8.** *Let  $T$  be a tree with vertex  $v$ , and let  $X, X' \in \{A, B, C, D, E\}$  with  $X < X'$ . We have  $\text{opt}_{X'}(T, v) \leq \text{opt}_X(T, v) \leq \text{opt}_{X'}(T, v) + 1$ .*

*Proof.* The first inequality follows immediately by noticing that the conditions for types  $A$  to  $E$  are increasingly constrained. Moreover, all types are feasible by considering  $L = V(T)$ , so  $\text{opt}_X(T, v)$  is always well-defined. This implies that the first inequality is true.

For the second inequality, assume we have  $X' \in \{A, B\}$  and  $L$  is an optimal solution of type  $X'$ . Then,  $L \cup \{v\}$  is a solution of type  $E$ . If  $X' \in \{C, D\}$ , let  $L$  be an optimal solution of type  $X'$ . If  $v \notin L$  (then  $X' = C$ ), then again  $L \cup \{v\}$  is a solution of type  $E$ . If  $v \in L$ , possibly some (unique) vertex  $w$  of  $T$  satisfies  $N(w) \cap L = \{v\}$ . Then,  $L \cup \{w\}$  is a solution of type  $E$ . Thus,  $\text{opt}_X(T, v) \leq \text{opt}_E(T, v) \leq \text{opt}_{X'}(T, v) + 1$ . In all cases, we have  $\text{opt}_X(T, v) \leq \text{opt}_E(T, v) \leq \text{opt}_{X'}(T, v) + 1$ , as claimed.  $\square$






					
	$T_A$	$T_B$	$T_C$	$T_D$	$T_E$
$opt_A$	0	1	2	1	3
$opt_B$	1	1	2	1	3
$opt_C$	1	2	2	1	3
$opt_D$	1	2	3	1	3
$opt_E$	1	2	3	2	3

Table 9.2: The five rooted tree gadgets with their values of the five  $opt_X$  functions.

By Lemma 9.8, for any rooted tree  $(T, v)$ , the five values  $opt_X(T, v)$  (for  $X \in \{A, B, C, D, E\}$ ) differ by at most one, and we know that they increase along with  $X$ :  $opt_A(T, v) \leq opt_B(T, v) \leq \dots \leq opt_E(T, v)$ . Hence, for any rooted tree  $(T, v)$ , there is one value  $X \in \{A, B, C, D, E\}$  so that, whenever  $X' \leq X$ , we have  $opt_{X'}(T, v) = opt_X(T, v)$ , and whenever  $X' > X$ , we have  $opt_{X'}(T, v) = opt_X(T, v) + 1$ . Thus, we can introduce the following definition, which partitions the set of rooted trees according to one of five such possible behaviors.

**Definition 9.2** (Rooted tree classes). For  $X \in \{A, B, C, D, E\}$ , we define the class  $\mathcal{T}_X$  as the set of pairs  $(T, v)$  such that  $T$  is a tree with root  $v$ , and there is an integer  $k$  with  $opt_{X'}(T, v) = k$  whenever  $X' \leq X$ , and  $opt_{X'}(T, v) = k + 1$  whenever  $X' > X$  (note that  $k$  may differ across the trees in  $\mathcal{T}_X$ ).

Note that for the trees in  $\mathcal{T}_E$ , all five types of optimal solutions have the same size. It is clear by Lemma 9.8 that the set of all possible pairs  $(T, v)$  where  $T$  is a tree and  $v$  a vertex of  $T$ , can be partitioned into the five classes  $\mathcal{T}_X$ , where  $X \in \{A, B, C, D, E\}$ . Moreover, it is not difficult to construct small rooted trees for each of these five classes. We provide five such small rooted trees in Table 9.2, each in a different class.

**Definition 9.3** (Rooted tree gadgets). For  $X \in \{A, B, C, D, E\}$ , let  $(T_X, v_X)$  be the rooted tree of class  $\mathcal{T}_X$  from Table 9.2, and let  $k_X = opt_X(T_X, v_X)$ .

We are now ready to present our first reduction rule.

**Reduction Rule 9.2.** Let  $(G, k)$  be an instance of LD-CODE such that  $G$  can be obtained from a graph  $G'$  and a tree  $T$ , by identifying a vertex  $v$  of  $T$  with a vertex  $w$  of  $G'$ , such that  $(T, v)$  is in class  $\mathcal{T}_X$  with  $X \in \{A, B, C, D, E\}$ , and with  $opt_X(T, v) = t$ . Then, we remove all vertices of  $T \setminus \{v\}$  from  $G$  (this results in  $G'$ ), we consider  $G$  and a copy of the rooted tree gadget  $(T_X, v_X)$ , and we identify  $v_X$  with  $v$ , to obtain  $G''$ . The reduced instance is  $(G'', k - t + k_X)$ .

**Lemma 9.9.** For an instance  $(G, k)$  of LD-CODE, Reduction Rule 9.2 can be applied exhaustively in time  $\mathcal{O}(n^2)$ , and for each application,  $(G, k)$  is a YES-instance of LD-CODE if and only if  $(G'', k - t + k_X)$  is a YES-instance of LD-CODE.

*Proof.* To apply Reduction Rule 9.2, one can find all vertices of degree 1 in  $G$ , mark them, and iteratively mark degree 1 vertices in the subgraph induced by unmarked vertices. Note that a connected subset  $S$  of marked vertices induces a tree with a single vertex  $v_S$  adjacent to a non-marked vertex  $w$ . Moreover, at the end of the marking process, the subgraph induced by non-marked vertices has minimum degree 2. This process can be done in  $\mathcal{O}(n^2)$  time.

We form the set of candidate trees  $T$  by taking the union  $S_w$  of all marked subsets that are connected to some common unmarked vertex  $w$ , together with  $w$ . Note that all these candidate trees are vertex-disjoint. Again, this can be done in  $\mathcal{O}(n^2)$  time. Let  $(T, v)$  be such a rooted tree, and let  $G'$  be the subgraph induced by  $V(G) \setminus S_v$ . Then, we apply the linear-time dynamic programming algorithm of Proposition 9.4 to  $(T, v)$  to determine the class  $\mathcal{T}_X$  to which  $(T, v)$  belongs to, for

some  $X \in \{A, B, C, D, E\}$ , as well as  $\text{opt}_X(T, v) = t$  for each  $X \in \{A, B, C, D, E\}$ . Constructing  $(G'', k - t + k_X)$  by replacing  $T$  by  $T_X$  in  $G$  can then be done in time  $\mathcal{O}(|V(T)|)$ . Thus, overall applying the reduction rule to all trees takes  $\mathcal{O}(n^2)$  time.

Now, we prove the second part of the statement. Assume we have applied Reduction Rule 9.2 to a single tree  $(T, v)$  to  $G$ , that  $(G, k)$  is a YES-instance of LD-CODE, and let  $L$  be an optimal locating-dominating set of  $G$  of order at most  $k$ . Assume that  $L \cap V(T)$  is of type  $X'$  for  $X' \in \{A, B, C, D, E\}$  with respect to  $(T, v)$ . Notice that the only interactions between the vertices in  $V(T)$  and  $V(G')$  are through the cut-vertex  $v$ . If  $X' \leq X$ , then, as  $L$  is optimal, we have  $|L \cap V(T)| = t$ ; if  $X' > X$ , then by Lemma 9.8, we have  $|L \cap V(T)| = t + 1$ . Now, we obtain a solution of  $G''$  from  $L' = L \cap V(G')$  by adding an optimal solution  $S_{X'}$  of type  $X'$  of  $(T_X, v_X)$  to  $L'$ . This has size  $k_X$  if  $X' \leq X$  and  $k_X + 1$  if  $X' > X$ . As  $S_{X'}$  behaves the same as  $L \cap V(T)$  with respect to  $L'$ ,  $L$  is a valid locating-dominating set of  $G''$ . Its order is either  $|L| - (t + 1) + k_X + 1$  if  $X' > X$ , or  $|L| - t + k_X$  if  $X \leq X'$ , which in both cases is at most  $k - t + k_X$ , thus  $(G'', k - t + k_X)$  is a YES-instance.

The proof of the converse follows by the same argument: essentially, the trees  $(T, v)$  and  $(T_X, v_X)$  are interchangeable.  $\square$

We now use Lemma 9.9 to apply Reduction Rule 9.2 exhaustively in time  $\mathcal{O}(n^2)$ , before proceeding to the second part of the algorithm.

**Step 2 – Handling the subgraphs corresponding to edges of  $\tilde{G}$ .** In the second step of our algorithm, we construct tree gadgets similar to those defined in Step 1, but with *two* distinguished vertices. We thus extend the above terminology to this setting. Let  $(T, v_1, v_2)$  be a tree  $T$  with two distinguished vertices  $v_1, v_2$ . We define types of trees as in the previous subsection. To do so, we first define the types for each of  $v_1$  and  $v_2$ , and then we combine them. For  $X \in \{A, B, C, D, E\}$ , a subset  $L$  of  $V(T)$  is of *type*  $(X, -)$  with respect to  $v_1$  if:

- Type  $(A, -)$ :  $L$  dominates all vertices of  $V(T) \setminus \{v_1, v_2\}$  and any two vertices of  $V(T) \setminus (L \cup \{v_1, v_2\})$  are located.
- Type  $(B, -)$ :  $L$  dominates all vertices of  $V(T) \setminus \{v_2\}$  and any two vertices of  $V(T) \setminus (L \cup \{v_1, v_2\})$  are located;
- Type  $(C, -)$ :  $L$  dominates all vertices of  $V(T) \setminus \{v_2\}$  and any two vertices of  $V(T) \setminus (L \cup \{v_2\})$  are located;
- Type  $(D, -)$ :  $L$  dominates all vertices of  $V(T) \setminus \{v_2\}$ ,  $v_1 \in L$ , and any two vertices of  $V(T) \setminus (L \cup \{v_2\})$  are located;
- Type  $(E, -)$ :  $L$  dominates all vertices of  $V(T) \setminus \{v_2\}$ ,  $v_1 \in L$ , any two vertices of  $V(T) \setminus (L \cup \{v_2\})$  are located, and there is no vertex  $w$  of  $V(T) \setminus \{v_2\}$  with  $N(w) \cap L = \{v_1\}$ ;

Thus, the idea for a set  $L$  of type  $(X, -)$  with respect to  $v_1$  is that there are specific constraints for  $v_1$ , but no constraint for  $v_2$  (and the usual domination and location constraints for the other vertices). We say that  $L$  is of type  $(X, Y)$  with respect to  $(T, v_1, v_2)$  if  $L$  is of type  $(X, -)$  with respect to  $v_1$  and of type  $(Y, -)$  with respect to  $v_2$ . Thus, we have a total of 25 such possible types. We let  $\text{opt}_{X,Y}(T, v_1, v_2)$  be the smallest order of a subset  $L$  of  $T$  of type  $(X, Y)$ . Here as well, there is a linear-time dynamic programming algorithm that, for any tree  $T$  and two vertices  $v_1, v_2$ , conjointly computes  $\text{opt}_{X,Y}(T, v_1, v_2)$  for all  $X, Y \in \{A, B, C, D, E\}$ .

**Proposition 9.5.** *There is a linear-time dynamic programming algorithm that, for any tree  $T$  and two vertices  $v_1, v_2$ , conjointly computes  $\text{opt}_{X,Y}(T, v_1, v_2)$  for all  $X, Y \in \{A, B, C, D, E\}$ .*

*Proof.* The dynamic programming is similar to that of Proposition 9.4. We root  $T$  at  $v_1$ . The main difference is that extra care must be taken around  $v_2$  to only keep the solutions of type  $(-, Y)$  for the subtrees that contain  $v_2$ .  $\square$

Next, similarly to Lemma 9.8, we show that for a triple  $(T, v_1, v_2)$ , the optimal solution values for different types, may vary by at most 2.



**Lemma 9.10.** *Let  $T$  be a tree with vertices  $v_1, v_2$  of  $T$ , and let  $X, X', Y, Y' \in \{A, B, C, D, E\}$  with  $X \leq X'$  and  $Y \leq Y'$ . If  $X = X'$  and  $Y < Y'$  or  $X < X'$  and  $Y = Y'$ , we have  $\text{opt}_{X,Y}(T, v_1, v_2) \leq \text{opt}_{X',Y'}(T, v_1, v_2) \leq \text{opt}_{X,Y}(T, v_1, v_2) + 1$ . If  $X < X'$  and  $Y < Y'$ , we have  $\text{opt}_{X,Y}(T, v_1, v_2) \leq \text{opt}_{X',Y'}(T, v_1, v_2) \leq \text{opt}_{X,Y}(T, v_1, v_2) + 2$ .*

*Proof.* The proof is similar to that of Lemma 9.8. As before, the lower bound is clear since the conditions for type  $(X', Y')$  are more constrained than those for  $(X, Y)$ . For the upper bound, assume first that  $X = X'$  or  $Y = Y'$  (without loss of generality,  $Y = Y'$ ). Then, as before, given a solution  $S$  of type  $(X, Y)$ , we can add to  $S$  a single vertex in the closed neighborhood of  $v_1$  to obtain a solution of type  $(E, Y)$ . Similarly, if both  $X < X'$  and  $Y < Y'$ , we can add to  $S$  a vertex to the closed neighborhoods of each of  $v_1$  and  $v_2$  and obtain a solution of type  $(E, E)$ .  $\square$

By Lemma 9.10, we can introduce the following definition. Note that for a triple  $(T, v_1, v_2)$ , the minimum value of  $\text{opt}_{X,Y}(T, v_1, v_2)$  over all  $X, Y \in \{A, B, C, D, E\}$  is reached for  $X = Y = A$  (and the maximum, for  $X = Y = E$ ).

**Definition 9.4** (Doubly rooted tree classes). For a function  $g : \{A, B, C, D, E\}^2 \rightarrow \{0, 1, 2\}$ , we define the class  $\mathcal{T}_g$  as the set of triples  $(T, v_1, v_2)$  such that  $T$  is a tree,  $v_1, v_2$  are two of its vertices, and for every  $(X, Y) \in \{A, B, C, D, E\}^2$ , we have  $\text{opt}_{X,Y}(T, v_1, v_2) - \text{opt}_{A,A}(T, v_1, v_2) = g(X, Y)$ .

There are at most  $3^{25}$  possible classes of triples, since that is the number of possible functions  $g$ . By Lemma 9.10, these classes define a partition of all sets of triples  $(T, v_1, v_2)$ .<sup>2</sup> We next define suitable tree gadgets for the above classes of triples. As there is a very large number of possible types of triples, we cannot give a concrete definition of such a gadget, but rather, an existential one.

**Definition 9.5** (Doubly-rooted tree gadgets). For a function  $g : \{A, B, C, D, E\}^2 \rightarrow \{0, 1, 2\}$ , let  $(T_g, v_X, v_Y)$  be the *smallest* tree in  $\mathcal{T}_g$  (if such a tree exists), and let  $k_g = \text{opt}_{A,A}(T_g, v_X, v_Y)$ .

We are now ready to present our second reduction rule, which replaces certain triples  $(T, v_1, v_2)$  by gadgets from Definition 9.5.

**Reduction Rule 9.3.** *Let  $(G, k)$  be an instance of LD-CODE, with:*

- *an induced subgraph  $G'$  of  $G$  with two vertices  $w_1, w_2$ ;*
- *a tree  $T$  with two vertices  $v_1, v_2$  with  $(T, v_1, v_2)$  in class  $\mathcal{T}_g$  with  $g : \{A, B, C, D, E\}^2 \rightarrow \{0, 1, 2\}$ , and with  $\text{opt}_{A,A}(T, v_1, v_2) = t$ ;*
- *$w_1, w_2$  do not belong to a hanging tree of  $G$ ;*
- *$G$  can be obtained by taking a copy of  $T$  and a copy of  $G'$  and identifying, respectively,  $v_1$  with  $w_1$ , and  $v_2$  with  $w_2$ .*

*Then, we remove all vertices of  $T \setminus \{v_1, v_2\}$  from  $G$  (that is, we compute  $G'$ ), we consider  $G'$  and a copy of doubly-rooted tree gadget  $(T_g, v_X, v_Y)$ , and we identify  $v_X$  with  $v_1$  and  $v_Y$  with  $v_2$ , to obtain  $G''$ . The reduced instance is  $(G'', k - t + k_g)$ .*

**Lemma 9.11.** *For an instance  $(G, k)$  of LD-CODE, Reduction Rule 9.3 can be applied to every tree component corresponding to an edge of  $\tilde{G}$  in time  $\mathcal{O}(n\text{fes})$ , and  $(G, k)$  is a YES-instance of LD-CODE if and only if  $(G'', k - t + k_g)$  is a YES-instance of LD-CODE.*

*Proof.* To apply the reduction rule as stated, we compute the multigraph  $\tilde{G}$  in time  $\mathcal{O}(n + \text{fes})$  using Proposition 9.3; by Observation 9.1, we can assume it has no loops and at most  $4\text{fes} - 3$  edges. Now, for every edge  $xy$  of  $\tilde{G}$ , we select the neighbor  $v_1$  of  $x$  and the neighbor  $v_2$  of  $y$  in  $G$ , that lie on the  $xy$ -path corresponding to the edge  $xy$  of  $\tilde{G}$ . The tree  $(T, v_1, v_2)$  is the subgraph of  $G$  corresponding to the edge  $xy$  of  $\tilde{G}$ , and  $G'$  is the subgraph of  $G$  obtained by removing  $T \setminus \{v_1, v_2\}$  from  $G$ . Computing all these subgraphs can be done in time  $\mathcal{O}(n\text{fes})$ . Computing the type  $\mathcal{T}_g$  of

<sup>2</sup>Some of these classes are actually empty, since, for example, by Lemma 9.10,  $\text{opt}_{X,Y}(T, v_1, v_2)$  increases with  $X$  and  $Y$ ; however, an empirical study shows that the number of nonempty classes is very large.

$(T, v_1, v_2)$  can be done in time  $\mathcal{O}(|T|)$  by Proposition 9.5, so overall, the running time of this step is  $\mathcal{O}(n\text{fes})$ .

Note that all the computed subgraphs are vertex-disjoint, we can thus now apply Reduction Rule 9.3 in time  $\mathcal{O}(|T|)$  to each tree  $T$ , noting that the gadget  $T_g$  is of constant order.

For the second part of the statement, the arguments are the same as those of Lemma 9.9: since they belong to the same class  $\mathcal{T}_g$ , the trees  $(T, v_1, v_2)$  and  $(T_g, v_X, v_Y)$  have the same behavior with respect to LD-CODE and the overall solution for the graph.  $\square$

**Completion of the proof.** By Lemma 9.9 and Lemma 9.11, we can apply Reduction Rule 9.2 and Reduction Rule 9.3 in time  $\mathcal{O}(n^2 + n\text{fes}) = \mathcal{O}(mn)$ . Since  $\tilde{G}$  has at most  $3\text{fes} - 2$  vertices and  $4\text{fes} - 3$  edges, we replaced each vertex of  $\tilde{G}$  by a constant-ordered tree-gadget from Definition 9.3 and each edge of  $\tilde{G}$  by a constant-ordered doubly-rooted tree gadget from Definition 9.5, the resulting graph has  $\mathcal{O}(\text{fes})$  vertices and edges, as claimed. The running time is  $2^{\mathcal{O}(\text{fes})} + n^{\mathcal{O}(1)}$  by first computing the kernel and then solving LD-CODE in time  $2^{\mathcal{O}(n)} = 2^{\mathcal{O}(\text{fes})}$  on that kernel. This completes the proof.

## 9.7 Conclusion

In this chapter, we presented several results that advance our understanding of the algorithmic complexity of LD-CODE, which we showed to have very interesting and rare parameterized complexities. Moreover, we believe the techniques used in this work can be applied to other identification problems to obtain relatively rare conditional lower bounds. The process of establishing such lower bounds boils down to designing *bit-representation gadgets* and *set-representation gadgets* for the problem in question. Apart from the broad question of designing such lower bounds for other identification problems, we mention an interesting problem left open by our work.

**Open Problem 9.1.** *Can the tight double-exponential lower bound for LD-CODE parameterized by treewidth/treedepth presented in this chapter be applied to the feedback vertex set number?*

The above question could also be studied for other related parameters. As for LD-CODE parameterized by vertex cover  $\text{vc}$  of the input graph, we do not know whether our  $2^{\mathcal{O}(\text{vc} \log \text{vc})}$ -time FPT-algorithm is optimal, or whether there exist single-exponential algorithms (with running time  $2^{\mathcal{O}(\text{vc})} \cdot n^{\mathcal{O}(1)}$ ). Another open question in this direction is the following.

**Open Problem 9.2.** *Can our FPT-algorithm for LD-CODE parameterized by vertex cover be generalized to when the problem is parameterized by the distance to cluster?*

Moreover, as our linear kernel for LD-CODE parameterized by feedback edge set is not explicit, it would also be nice to obtain a concrete kernel.



## Chapter 10

# Conclusion and perspectives

We close with a summary of our work in this thesis and outline some perspectives for the problems within the framework of the existing and ongoing research on identification problems on graphs.

## 10.1 Summary of our work

In this thesis, we have primarily studied four different types of separating sets combined with dominating and total-dominating sets in graphs. This gave rise to eight different types of X-codes for  $X \in \text{CODES} = \{\text{LD}, \text{LTD}, \text{ID}, \text{ITD}, \text{OD}, \text{OTD}, \text{FD}, \text{FTD}\}$ . Three of these codes, namely OD-, FD- and the FTD-code, have been introduced rather recently to the literature of identification problems. The study carried out in this thesis begins with a comparison of the eight X-numbers among each other. To carry out this comparison, we made use of a canonical reformulation of the problem of finding X-numbers of graphs to an equivalent problem of finding the covering number of a corresponding hypergraph (called the X-hypergraph). Thereafter, the hyperedges of these X-hypergraphs provide a natural playing field for pairwise comparisons of these code numbers. We then made use of another tool, namely a generalized reformulation — similar to those used in [42] — of Bondy's Theorem [29]. With this tool, we showed that it is also possible to reverse the orders of these comparisons (albeit with a multiplicative factor). This provides us with a comprehensive picture of all the eight codes studied in this thesis and how their respective code numbers compare with each another. We also provided some general upper and lower bounds on these codes. Moreover, for each  $X \in \text{CODES}$ , we provided a generic description of all graphs whose X-number achieves this general lower bounds (which are already known from the literature to be logarithmic in the order of the graph).

As far as the study related to each code is concerned, the structure of this thesis can be broadly divided into two parts:

- Part I. Structural results concerning upper bounds and lower bounds on the code numbers of graphs with a special focus on studying these codes on graphs that belong to some well-known graph families.
- Part II. Algorithmic aspects regarding X-CODE, mainly with a focus on  $X \in \{\text{LD}, \text{OD}, \text{FD}, \text{FTD}\}$ . For the last three codes, we focused on determining the hardness of the problem and for  $X = \text{LD}$  (already known to be NP-complete from the literature), we ventured into some of the parameterized complexity aspects of LD-CODE.

### 10.1.1 Part I. Structural aspects

As far as structural aspects of the problems are concerned, we studied each type of separating set in a separate chapter and tried to analyze the respective code numbers both in general and on graphs of some special graph families.

In Chapter 4, we looked at the separation property of location and studied it in combination with domination (that is, the LD-codes) and total-domination (that is, the LTD-codes). Our study of LD-codes in the chapter was over the graph families of block graphs and subcubic graphs. For block graphs, we found general tight upper and lower bounds of their LD-numbers. In addition, we also verified for block graphs the following  $n$ -half upper bound Conjecture 2.2 and proved it to be both true and tight.

**Conjecture 2.2** ([112, 95]) *Let  $G$  be a twin-free and isolate-free graph on  $n$  vertices. Then, we have*

$$\gamma^{\text{LD}}(G) \leq \frac{n}{2}.$$

On subcubic graphs, we continued our study of the  $n$ -half upper bound, that is, Conjecture 2.2 and proved it to be also true for this graph class. Moreover, we expanded the scope of this conjecture even further for subcubic graphs by proving that the conjecture is true even when subcubic graphs are allowed to have closed twins and open twins of degree 3 (except when the graph is either  $K_4$  and  $K_{3,3}$ ). This also answers positively to questions from the literature posed in [95] about the veracity of the said conjecture for cubic graphs with twins. However, we showed that the conjecture is not true for all regular graphs with twins in general as we showed that the bound fails for  $r$ -regular graphs with  $r \geq 4$ .

In the second part of Chapter 4, we studied locating total-dominating codes and found bounds on LTD-numbers of graphs from several graph families like cobipartite graphs, split graphs, block graphs and subcubic graphs. More precisely, we studied Conjecture 2.3, restated hereby, for all graphs from the said graph families and proved it to be true and tight.

**Conjecture 2.3** ([96]) *Every twin-free and isolate-free graph  $G$  of order  $n$  satisfies*

$$\gamma^{\text{LTD}}(G) \leq \frac{2}{3}n.$$

Our results for some of the graph families are, in fact, even stronger than the conjecture. For example, instead of the conjectured upper bound of two-thirds the order of the graph ( $\frac{2n}{3}$ ), for cobipartite graphs we showed that this bound falls to  $n$ -half and for split graphs it falls to being strictly less than  $\frac{2n}{3}$ . Furthermore, we showed that this upper bound (conjectured to hold for twin-free graphs) also holds for subcubic graphs even when we allow the graphs to have twins.

In Chapter 5, we studied identifying codes of graphs on different graph families. Here too, we found tight upper and lower bounds on the ID-numbers of block graphs. In fact, the upper bound we studied for ID-numbers of block graphs is the following conjecture made in [7] in terms of the number of blocks of a block graph.

**Conjecture 5.1** ([7]). *The ID-number of a closed-twin-free block graph is bounded above by the number of blocks in the graph.*

Thereafter, in the same chapter, we moved on to studying ID-numbers of graphs in terms of their maximum degree and order. This is motivated by the following conjecture which we proved to be true for trees and triangle-free graphs.

**Conjecture 5.2.** ([99]). *There exists a constant  $c$  such that for every connected identifiable graph of order  $n \geq 2$  and of maximum degree  $\Delta \geq 2$ ,*

$$\gamma^{\text{ID}}(G) \leq \left( \frac{\Delta - 1}{\Delta} \right) n + c.$$

We first considered Conjecture 5.2 for trees. With the conjecture already being true for paths (with  $\Delta = 2$ ) from the literature, we focused mainly on the case when  $\Delta \geq 3$ . To that end, we showed that except for a small list  $\mathcal{T}_\Delta$  of exceptional graphs, the conjecture holds for trees with  $\Delta \geq 3$  with the constant  $c = 0$  in the conjecture. Moreover, we characterized the set  $\mathcal{T}_\Delta$  of exceptional trees for which  $c > 0$  and proved that for  $\Delta = 3$ , the set  $\mathcal{T}_\Delta$  is the set of 12 trees of maximum degree 3

and diameter at most 6 depicted in Figure 5.2; and  $\mathcal{T}_\Delta = \{K_{1,\Delta}\}$  for  $\Delta \geq 4$ . Along the way, we also proved an interesting interplay between the ID-number and domination number of a tree and showed that their sum cannot be larger than the order of the tree.

We next moved on to studying Conjecture 5.2 for triangle-free graphs and again proved it to be true there. Here too, we focused on the case with  $\Delta \geq 2$  (as the result with  $\Delta = 2$  is again known to be true from the literature). Our result on Conjecture 5.2 for trees was a starting point of our proof for the same conjecture for triangle-free graphs. Also, along the way, we proved a generalized version of Bondy's theorem [29] on induced subsets that we used as a tool in our proofs. We also used our main result for triangle-free graphs, to prove the upper bound  $\left(\frac{\Delta-1}{\Delta}\right)n + 1/\Delta + 4t$  for graphs that can be made triangle-free by the removal of  $t$  edges.

In Chapter 6, we studied the property of open separation combined both with domination (that is, OD-codes) and total-domination (that is, OTD-codes). In the first half of the chapter, we studied OTD-codes on certain graph families. To begin with, we again considered block graphs and provided tight upper and lower bounds on the OTD-numbers of block graphs. Our upper bound is a generalization of a result in [89] for half-graphs (of which  $P_2$  and  $P_4$  are also block graphs). In this chapter, we also looked at OTD-codes of cycles and proved an exact result for these code numbers in terms of the order of the cycle. We then went on to consider bounds on OTD-numbers of graphs in terms of their maximum degree and order. More precisely, we proved the following upper bound for  $C_4$ -free graphs.

**Theorem 6.5.** *For  $\Delta \geq 3$  a fixed integer, if  $G$  is an open-twin-free connected graph of order  $n \geq 5$  that contains no 4-cycles and satisfies  $\Delta(G) \leq \Delta$ , then*

$$\gamma^{\text{OTD}}(G) \leq \left(\frac{2\Delta - 1}{2\Delta}\right)n,$$

*except in one exceptional case when  $G$  is isomorphic to a subdivided  $\Delta$ -star, in which case*

$$\gamma^{\text{OTD}}(G) = \left(\frac{2\Delta}{2\Delta + 1}\right)n.$$

We first proved Theorem 6.5 for trees (also  $C_4$ -free) and then used the result on trees to prove the theorem for all  $C_4$ -free graphs.

In the second half of the same chapter we looked at OD-codes of graphs which were introduced to the literature of identification problems rather recently in [52]. We showed that OD-codes happen to be quite closely associated to OTD-codes in that, the two code numbers can differ by at most 1. This motivated us to study the OD-numbers (in comparison with the OTD-numbers already known from the literature) of some graph families like cliques, matchings and their disjoint unions, half-graphs and subdivided stars as subfamilies of bipartite graphs, some subfamilies of split graphs (which includes some thin suns) and headless spiders. Our results on the OD-numbers for all these graph families are exact. We also studied OD-codes from a polyhedral point of view providing the OD-polyhedra of cliques, matchings, and thick and thin headless spiders.

In Chapter 7, we again studied two codes introduced to the literature of identification problems rather recently in [53]. More precisely, we studied FD-codes (full separation combined with domination) and FTD-codes (full separation combined with total-domination) on graphs. As in the case of OD- and OTD-codes, we found that even FD- and FTD-codes are in close association with one another in that, the respective code numbers differ by at most 1. To that end, we studied these two code numbers and compared them on several graph families like paths, cycles, half-graphs and headless spiders. Here again, all our results on FD- and FTD-code numbers are exact.

## 10.1.2 Part II. Algorithmic aspects

Chapters 8 and 9 are dedicated to the study of the algorithmic and computational complexity aspects of the identification problems that we considered in this thesis.

In Chapter 8, we studied the computational complexity of OD-CODE, FD-CODE and FTD-CODE, which are the decision versions of the problems of finding the OD-numbers, FD-numbers and the FTD-numbers, respectively, of graphs. We proved that all three problems are NP-complete. Apart from these, we also considered the problem of deciding if the OD-number and the OTD-number (respectively, the FD-number and the FTD-number) of a graph are equal or not. We showed that, even with a difference of at most 1 between each of the pairs, it is NP-hard to decide in general if the numbers are equal or not.

In Chapter 9, we looked at the parameterized complexity aspects of LD-CODE. In this chapter, we studied the problem LD-CODE parameterized by both natural parameters like solution size, order of the graph and also structural parameters like treewidth, vertex cover number, twin cover number, distance to clique, neighborhood diversity and feedback edge set number.

To begin with, we studied LD-CODE, say, with an input graph on  $n$  vertices, and parameterized by the natural parameter of solution size, say  $k$ . We proved that, under the ETH the problem does not admit a running time of  $2^{o(k^2)} \cdot n^{\mathcal{O}(1)}$ . This essentially ensures a tight running time of the order of  $2^{\mathcal{O}(k^2)} \cdot n^{\mathcal{O}(1)}$  for the problem (the upper bound for this running time comes from the fact that LD-CODE has a kernel of order  $\mathcal{O}(2^k)$  — the latter result being available from preliminary results in [186]). This lower bound improves on two previously existing lower bounds of  $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$  under  $W[2] \neq \text{FPT}$  in [19] and  $2^{o(k \log k)}$  under the ETH in [37]. We then examined the incompressibility of LD-CODE and proved that, unless  $\text{NP} \subseteq \text{coNP}/\text{poly}$ , the problem does not admit a polynomial compression of order  $\mathcal{O}(n^{2-\epsilon})$  for any  $\epsilon > 0$ .

We then continued to explore LD-CODE parameterized by structural parameters. We began with treewidth (tw) and showed that, under the ETH, the problem does not admit a running time of  $2^{2^{o(\text{tw})}} \cdot \text{poly}(n)$ . This result adds the problem to the small list of problems from the class NP which require at least a double-exponential running time parameterized by treewidth (the other three being METRIC DIMENSION, STRONG METRIC DIMENSION and GEODETIC SET considered in [93]). We also showed that this lower bound on the running time is tight by providing a dynamic programming algorithm that solves the problem in a running time of  $2^{2^{\mathcal{O}(\text{tw})}} \cdot n^{\mathcal{O}(1)}$ .

We then looked at LD-CODE with vertex-cover number (vc) as a parameter and provided a slightly super-exponential algorithm by dynamic programming for the problem with a running time of  $2^{\mathcal{O}(\text{vc} \log \text{vc})} \cdot n^{\mathcal{O}(1)}$ . We showed that this result can be extended to hold even with other parameters like distance to clique and twin-cover number. Thereafter, we also went on to show that LD-CODE has a linear kernel when parameterized by neighborhood diversity. Finally, we also showed that such linear kernels to the problem also exist when parameterized by the feedback edge set number.

## 10.2 Perspectives and future research

We finally provide a perspective of the work carried out in this thesis in relation to other similar and related research being carried out in the domain of identification problems. This helps to highlight how the problems studied in this thesis fit into a broader and more general approach to identification problems on graphs. Moreover, it leads to an exchange of similar questions that can be explored as future work and, perhaps also, proof techniques that can be applied across the domain of identification problems. To begin with, in the next subsection, we provide a selected list of identification problems on graphs that we think are closely related — in both objective, applications and scope of research — to those studied in this thesis. Needless to say, such closely related problems naturally pose the question of

*“How do the results — such as bounds, algorithms, complexities etc. — that we have studied in this thesis, present themselves in the context of other similar problems; and vice-versa?”*

### 10.2.1 Related identification problems on graphs

All the problems presented in this subsection also bear the same objective of distinguishing pairs of vertices and minimizing the solution size. Hence, most questions explored for the problems considered in this thesis also carry over to the following set of problems.

**Definition 10.1** (Resolving set [185]). Given a graph  $G$ , a set  $S \subseteq V(G)$  is called a *resolving set* of  $G$  if for any two distinct vertices  $u, v \in V(G)$ , there exists a vertex  $x \in S$  such that  $d_G(u, x) \neq d_G(v, x)$ .

Notice that, for any vertex  $v$  in a resolving set  $S$ , we have  $d_G(v, v) = 0$  and hence,  $v$  is always separated from all other vertices by  $S$ . Thus, resolving sets can really be seen as the non-local version of LD-codes with the only difference being that one does not (classically speaking) require a resolving set to be a (generalized) dominating set. Resolving sets also happen to be one of the more well-studied identification problems in graphs. However, such varieties of problems have also been studied, for example, in the concept of *resolving dominating sets* [34].

**Definition 10.2** (Redundant OLD-set [192]). Given a graph  $G$ , a set  $S \subseteq V(G)$  is called a *redundant OLD-set* of  $G$  if  $|N_G(v) \cap S| \geq 2$  for all  $v \in V(G)$  and  $|(N_G(u) \cap S) \Delta (N_G(v) \cap S)| \geq 2$ , for all distinct  $u, v \in V(G)$ .

Another equivalent way to think of a redundant OLD-set  $S$  of a graph  $G$  is that  $S$  is an OTD-code of  $G$  such that, for all  $v \in V(G)$ , the set  $S \setminus \{v\}$  is also an OTD-code of  $G$ . In Definition 10.2, the term “OLD” refers to the abbreviation from the name “open neighborhood-locating dominating” which was originally used by the authors in [190] instead of “open-separating total-dominating” (or OTD). We recall here that while describing how to tackle fault type 2 in Section 1.1.3 which finally led to our OTD-code, we had mentioned that it was important to seize an intruder at a code vertex, say  $w$ , of a graph  $G$  where they have disabled the detector before they moved to other vertices of  $G$ . Since Definition 10.2 implies that the set  $S \setminus \{w\}$  is still an OTD-code of  $G$ , a redundant OLD-code provides a model for the monitoring system to be still functioning as an OTD-code even if the intruder disables the detector at  $w$  and moves to other vertices of  $G$ .

**Definition 10.3** (Detector OLD-set [192]). Given a graph  $G$ , a set  $S \subseteq V(G)$  is called a *detector OLD-set* of  $G$  if  $|N_G(v) \cap S| \geq 2$  for all  $v \in V(G)$  and either  $|((N_G(u) \setminus N(v)) \cap S)| \geq 2$  or  $|((N_G(v) \setminus N(u)) \cap S)| \geq 2$ , for all distinct  $u, v \in V(G)$ .

A detector OLD-set is a stronger notion than a redundant OLD-set, whereby the former set models a monitoring system that can withstand not only (at most) one dysfunctional detector but also when a detector produces false positive signals to indicate an intruder.

Exactly similar concepts of *redundant LD-set* and *detector LD-set* (also called *fault-tolerant LD-set* in [188]) as above were defined by Slater in [187, 188]; and the concept of *fault-tolerant resolving sets* were developed by Hernando et al. in [131]. Similar fault-tackling mechanisms have also been studied using identifying codes under the name of strongly  $(t, \leq l)$ -identifying codes in [133]. Jean and Seo in [142] also defined the following concept which deals with withstanding faults with detectors that may be both false positives and false negatives.

**Definition 10.4** (Error-correcting LD-set [142]). Given a graph  $G$ , a set  $S \subseteq V(G)$  is called a *error-correcting LD-set*, or an *ERR:LD-set* for short, of  $G$  if  $|N_G(v) \cap S| \geq 3$  for all  $v \in V(G)$  and  $|(N_G(u) \cap S) \Delta (N_G(v) \cap S) \setminus \{u, v\}| \geq 3 - |\{u, v\} \cap S|$  for all distinct  $u, v \in V(G)$ .

We note here that an ERR:LD-set is a stronger notion than that of an FD-code and an FTD-code that we have defined here. Similar notions of *error-correcting OLD-sets* and *error-correcting ID-sets* have also been defined in [144, 143, 193]. Some even further stronger concepts of LD-codes called *self-locating-dominating codes* and *solid-locating-dominating codes* were defined in [146] to even locate an irregularity in a monitoring if such an irregularity were to happen. Along similar lines, we also have the concept of *self-identifying codes* defined in [147].

Apart from the above stronger notions of X-codes to aid in fault-detection, the definition of each X-code can be restricted to either a *local X-code* or a *non-local X-code*. These are defined in detail in Section 10.2.2. Another fairly widely studied related concept is that of a test cover defined below.



**Definition 10.5** (Test cover [30, 166]). Let  $(E, \mathcal{A})$  be a tuple where  $E$  is a set each of whose elements is called an *entity* and  $\mathcal{A}$  is a set of subsets of  $E$ . Moreover, each  $A \in \mathcal{A}$  is called an *attribute*. Then a subset  $\mathcal{T} \subseteq \mathcal{A}$  is called a *test cover* if, for any pair of distinct entities  $e, f \in E$ , there exists an attribute in  $\mathcal{T}$  to which exactly one of  $e$  and  $f$  belongs.

For even further codes and related concepts, we draw the attention of the reader to the online bibliography [141] maintained by Jean and Lobstein.

## 10.2.2 Further classification of codes and potential future problems

In this subsection, we show that the eight problems of X-codes that we have studied in this thesis can be fit into a much larger scheme. Trying out even further combinations of the four separating properties with the properties of domination and total-domination lead to the birth of even further interesting problems that may find real-life applications depending on the need.

To demonstrate this idea, we describe the eight problems in this thesis from an even more general point of view. Let  $S$  be a vertex subset of a graph  $G$ . Let  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$ . Then, using Remark 2.21, we notice that all the A-separators and B-neighborhoods that we have encountered so far have been defined in terms of  $N_C(G; v)$  and  $N_O(G; v)$  for  $v \in V(G)$  and depending on  $A \in \text{SEP-TYPE}$  and  $B \in \text{NBD-TYPE}$ . However, in the most abstract sense, for any two distinct  $u, v \in V(G)$ , one could also define a  $(u, v)$ -separator or a  $v$ -neighborhood to be any subset of  $V(G)$ . Let us look at the following example about separating sets.

Let  $\mathcal{U} = \{U(G; u, v) \subseteq V(G) : u, v \in V(G), u \neq v\}$  be any pre-defined set of subsets of  $V(G)$ . Then, we say that  $S$  is a  $\mathcal{U}$ -separating set of  $G$  if and only if  $U(G; u, v) \cap S \neq \emptyset$  for each pair of distinct  $u, v \in V(G)$ . This provides more room to define even further potential identification problems. For example, let us say that we would like to define a new kind of separating set called a *local open-separating set* whereby we call a non-empty vertex subset  $S$  of  $G$  to be a local open-separating set of  $G$  if  $S$  open-separates all *adjacent* vertices of  $G$ . We claim that such a local open-separating set is equivalent to a  $\mathcal{U}$ -separating set for some appropriate set  $\mathcal{U}$ . This claim is proven in the following proposition.

**Proposition 10.1.** *Let  $G$  be a graph and let  $\mathcal{U} = \{U(G; u, v) \subseteq V(G) : u, v \in V(G), u \neq v\}$  be such that  $U(G; u, v) = V(G)$  for all non-adjacent vertices  $u, v \in V(G)$  and  $U(G; u, v) = \Delta_O(G; u, v)$  for all adjacent vertices  $u, v \in V(G)$ . Then, any vertex subset  $S$  of  $G$  is a local open-separating set if and only if  $S$  is a  $\mathcal{U}$ -separating set.*

In the same manner as in Proposition 10.1, a separating set of any type  $A \in \text{SEP-TYPE}$  can be extended to a local and a non-local version by modifying the separators  $\Delta_A(G; u, v)$  to be  $\Delta_A(G; u, v) = V(G)$  for non-adjacent vertices  $u, v \in V(G)$  and  $\Delta_A(G; u, v) = V(G)$  for adjacent vertices  $u, v \in V(G)$ , respectively. These local and non-local versions of the separating sets of all types  $A \in \text{SEP-TYPE}$  are defined in rows 8 to 11 in Table 10.1.

By the same reasoning as in the case of separating sets, whenever we do not require a vertex subset of a graph to have either the property of domination or total domination, we simply update the neighborhoods to be  $V(G)$ . This is also reflected in row 1 of Table 10.1 where we only present a vertex subset; and from rows 4 to 11 of Table 10.1 where we only present the separating sets.

As a result, these local and non-local separating sets then also give rise to local and non-local versions of X-codes for all  $X \in \text{CODES}$ . These local and non-local X-codes for  $X \in \text{CODES}$  are presented in rows 20 to 27 in Table 10.1. As a naming convention, the local and non-local versions of any X-code, for  $X \in \text{CODES}$ , are named *local X-code* (or *LX-code* for short) and *non-local X-code* (or *NLX-code* for short), respectively.

	Subset name	Abbvr.	$\Delta_B^1(G)$	$\Delta_B^{2+}(G)$	$N_B(G)$
1	vertex subset		$\{V(G)\}$	$\{V(G)\}$	$\{V(G)\}$
2	dominating set	D-set	$\{V(G)\}$	$\{V(G)\}$	$N_C(G)$
3	total-dominating set	TD-set	$\{V(G)\}$	$\{V(G)\}$	$N_O(G)$
4	closed-separating set [149]	CS-set	$\Delta_C^1(G)$	$\Delta_C^{2+}(G)$	$\{V(G)\}$
5	open-separating set [190]	OS-set	$\Delta_O^1(G)$	$\Delta_O^{2+}(G)$	$\{V(G)\}$
6	locating set [184, 121]	LS-set	$\Delta_O^1(G)$	$\Delta_C^{2+}(G)$	$\{V(G)\}$
7	full-separating set [53]	FS-set	$\Delta_C^1(G)$	$\Delta_O^{2+}(G)$	$\{V(G)\}$
8	local closed-separating set [132]	LCS-set	$\Delta_C^1(G)$	$\{V(G)\}$	$\{V(G)\}$
9	local open-separating set [NOT STUDIED]	LOS-set	$\Delta_O^1(G)$	$\{V(G)\}$	$\{V(G)\}$
10	non-local closed-separating set [NOT STUDIED]	NLCS-set	$\{V(G)\}$	$\Delta_C^{2+}(G)$	$\{V(G)\}$
11	non-local open-separating set [NOT STUDIED]	NLOS-set	$\{V(G)\}$	$\Delta_O^{2+}(G)$	$\{V(G)\}$
12	identifying code [149]	ID-code	$\Delta_C^1(G)$	$\Delta_C^{2+}(G)$	$N_C(G)$
13	open-separating dominating code [52]	OD-code	$\Delta_O^1(G)$	$\Delta_O^{2+}(G)$	$N_C(G)$
14	locating dominating code [186]	LD-code	$\Delta_O^1(G)$	$\Delta_C^{2+}(G)$	$N_C(G)$
15	full-separating dominating code [53]	FD-code	$\Delta_C^1(G)$	$\Delta_O^{2+}(G)$	$N_C(G)$
16	identifying total-dominating code [123]	ITD-code	$\Delta_C^1(G)$	$\Delta_C^{2+}(G)$	$N_O(G)$
17	open-separating total-dominating code [190, 133]	OTD-code	$\Delta_O^1(G)$	$\Delta_O^{2+}(G)$	$N_O(G)$
18	locating total-dominating code [123]	LTD-code	$\Delta_O^1(G)$	$\Delta_C^{2+}(G)$	$N_O(G)$
19	full-separating total-dominating code [53]	FTD-code	$\Delta_C^1(G)$	$\Delta_O^{2+}(G)$	$N_O(G)$
20	local identifying code [132]	LID-code	$\Delta_C^1(G)$	$\{V(G)\}$	$N_C(G)$
21	local locating dominating code [132]	LLD-code	$\Delta_O^1(G)$	$\{V(G)\}$	$N_C(G)$
22	local identifying total-dominating code [NOT STUDIED]	LITD-code	$\Delta_C^1(G)$	$\{V(G)\}$	$N_O(G)$
23	local locating total-dominating code [NOT STUDIED]	LLTD-code	$\Delta_O^1(G)$	$\{V(G)\}$	$N_O(G)$
24	non-local identifying code [NOT STUDIED]	NLID-code	$\{V(G)\}$	$\Delta_C^{2+}(G)$	$N_C(G)$
25	non-local open-separating dominating code [NOT STUDIED]	NLOD-code	$\{V(G)\}$	$\Delta_O^{2+}(G)$	$N_C(G)$
26	non-local identifying total-dominating code [NOT STUDIED]	NLITD-code	$\{V(G)\}$	$\Delta_C^{2+}(G)$	$N_O(G)$
27	non-local open-separating total-dominating code [NOT STUDIED]	NLOTD-code	$\{V(G)\}$	$\Delta_O^{2+}(G)$	$N_O(G)$

Table 10.1: The table displays a list of different types of vertex subsets of a graph depending on which dominating and separating properties we use to define these subsets. Each dist-1 separator set has three options between  $\Delta_C^1(G)$ ,  $\Delta_O^1(G)$  and  $V(G)$ ; each dist-2+ separator set has three options between  $\Delta_C^{2+}(G)$ ,  $\Delta_O^{2+}(G)$  and  $V(G)$ ; and each neighborhood has three options between  $N_C(G)$ ,  $N_O(G)$  and  $V(G)$ . In total, therefore, there are 27 types of different vertex subsets defined in the table.

It can be noticed that in the local versions, an LID-code is the same as an LFD-code (hence, not in Table 10.1) and, therefore, an LITD-code is also the same as an LFTD-code (hence, not in Table 10.1); whereas an LLD-code is the same as an LOD-code (hence, not in Table 10.1) and, therefore, an LLTD-code is also the same as an LOTD-code (hence, not in Table 10.1). Similarly, in the non-local versions, an NLID-code is the same as an NLLD-code (hence, not in Table 10.1) and, therefore, an NLITD-code is the same as an NLLTD-code (hence, not in Table 10.1); whereas an NLOD-code is the same as an NLFD-code (hence, not in Table 10.1) and, therefore, an NLOTD-code is the same as an NLFTD-code (hence, not in Table 10.1). Thus, there are 4 less rows for each of the local and non-local versions of X-codes than there are for the original X-codes.

In line with such local versions of codes, recently in [132], Herva et al. have studied the local locating dominating and the locating identifying codes (see rows 20 and 21 of Table 10.1).

## 10.3 Concluding remarks

We conclude this thesis by highlighting the fact that the purpose of this work was to study identification problems both from a unifying point of view as well as individually. Since the scope of such problems is quite large, our goal was to choose a small subset of this research area. This subset of identification problems considered in this thesis consists of combining four separation properties, namely location, closed separation, open separation and full separation with the two most widely studied domination properties of (closed) domination and total-domination. This gave rise to eight types of codes which had been the focus of our study. The first three separation properties (of location, open separation and closed separation) had already been studied in the literature of identification problems. The fourth property (that of full separation) was a natural outcome of combining the previous three and completes the picture of the set of eight codes that we have tried to look at. This accounts for studying these codes from a unifying point of view.

As far as studying the codes individually is concerned, each separation type considered in our study has a chapter of the thesis dedicated to it. In each such dedicated chapter, we have studied the separation properties on specific graph families, a fair amount of which has been within the context of several important structural conjectures existing in the literature. Hence, one of the achievements of this thesis would be to contribute to affirming several conjectures and popular intuitive beliefs around the subject. We also studied some specific codes from the algorithmic point of view and achieved in providing bounds (sometimes tight) in the running time of the problem, thus establishing the computational challenges of these problems.

We hope that this work inspires the community to advance some of the ideas, techniques, questions and concepts undertaken and put forward during the course of this work.

# Bibliography

- [1] Agrawal, A., Kanesh, L., Saurabh, S., Tale, P.: Paths to trees and cacti. *Theoretical Computer Science* **860**, pp. 98-116 (2021).
- [2] Agrawal, A., Lokshitanov, D., Saurabh, S., Zehavi, M.: Split contraction: The untold story. *ACM Transactions on Computer Theory* **11**(3), pp. 1-22 (2019).
- [3] Archdeacon, D., Ellis-Monaghan, J., Fischer, D., Froncek, D., Lam, P.C.B., Seager, S., Wei, B., Yuster, R.: Some remarks on domination. *Journal of Graph Theory* **46**, pp. 207-210 (2004).
- [4] Argiroffo, G.R., Bianchi, S.M.: On the set covering polyhedron of circulant matrices. *Discrete Optimization* **6**, pp. 162-173 (2009).
- [5] Argiroffo G.R., Bianchi S.M., Wagler A.K.: Study of identifying code polyhedra for some families of split graphs. In: *International Symposium on Combinatorial Optimization*, Springer, pp. 13-25 (2014).
- [6] Argiroffo, G.R., Bianchi, S.M., Wagler, A. K.: Progress on the description of identifying code polyhedra for some families of split graphs. *Discrete Optimization* **22**, pp. 225-240 (2016).
- [7] Argiroffo, G.R., Bianchi, S.M., Lucarini Y., Wagler A.K.: On the identifying code number of block graphs. In: *Proceedings of ICGT 2018, Lyon, France* (2018).
- [8] Argiroffo, G.R., Bianchi, S.M., Lucarini Y., Wagler A.K.: Polyhedra associated with identifying codes in graphs. *Discrete Applied Mathematics* **245**, pp. 16-27 (2018).
- [9] Argiroffo, G.R., Bianchi, S.M., Lucarini, Y., Wagler, A.K.: The identifying code, the locating-dominating, the open locating-dominating and the locating total-dominating problems under some graph operations. *Electronic Notes in Theoretical Computer Science* **346**, pp. 135-145 (2019).
- [10] Argiroffo, G.R., Bianchi, S.M., Lucarini, Y., Wagler, A. K.: Linear-time algorithms for three domination-based separation problems in block graphs. *Discrete Applied Mathematics* **281**, pp. 6-41 (2020).
- [11] Argiroffo, G.R., Bianchi, S.M., Lucarini, Y., Wagler, A. K.: Polyhedra associated with locating-dominating, open locating-dominating and locating total-dominating sets in graphs. *Discrete Applied Mathematics* **322**, pp. 465-480 (2022).
- [12] Arkin, E.M., Banik, A., Carmi, P., Citovsky, G., Katz, M.J., Mitchell, J.S.B, Simakov, M.: Selecting and covering colored points. *Discrete Applied Mathematics* **250**, pp. 75-86 (2018).
- [13] Auger, D.: Minimal identifying codes in trees and planar graphs with large girth. *European Journal of Combinatorics* **31**(5), pp. 1372-1384 (2010).
- [14] Auger, D., Charon, I., Hudry, O., Lobstein, A.: Complexity results for identifying codes in planar graphs. *International Transactions in Operational Research* **17**(6), pp. 691-710 (2010).

- [15] Babai, L.: On the complexity of canonical labelling of strongly regular graphs. *SIAM Journal of Computing* **9**(1), pp. 212–216 (1980).
- [16] Balas, E., Ng, S.M.: On the set covering polytope: I. All the facets with coefficients in  $\{0,1,2\}$ . *Mathematical Programming* **43**, pp. 57–69 (1989).
- [17] Bandelt, H.J., Mulder, H.M.: Distance-hereditary graphs. *Journal of Combinatorial Theory, Series B* **41**(2), pp. 182–208 (1986).
- [18] Balbuena C., Foucaud F., Hansberg A.: Locating-Dominating Sets and Identifying Codes in Graphs of Girth at least 5. *The Electronic Journal of Combinatorics* **22**(2), P2.15 (2015).
- [19] Barbero, F., Isenmann, L., Thiebaut, J.: On the distance identifying set meta-problem and applications to the complexity of identifying problems on graphs. *Algorithmica* **82**(8), pp. 2243–2266 (2020).
- [20] Bearson L.W., Carroll, C.C.: On the design of minimum length fault tests for combinational circuits. *IEEE Transactions on Computers* **20**, pp. 1353–1356 (1971).
- [21] Bellitto, T., Brosse, C., Lévêque, B., Parreau, A.: Locating-dominating sets in local tournaments. *Discrete Applied Mathematics* **337**, pp. 14–24 (2023).
- [22] Berman, P., DasGupta, B., Kao, M.Y.: Tight approximability results for test set problems in bioinformatics. *Journal of Computer Systems and Sciences* **71**(2), pp. 145–162 (2005).
- [23] Bertrand, N.: Codes identifiants et codes localisateurs-dominateurs sur certains graphes. Master thesis, ENST, France, (2001).
- [24] Bertrand N., Charon I., Hudry O., Lobstein A.: Identifying and locating-dominating codes on chains and cycles. *European Journal of Combinatorics* **25**(7), pp. 969–987 (2004).
- [25] Bertrand N., Charon I., Hudry O., Lobstein A.: 1-Identifying codes on trees. *Australasian Journal of Combinatorics* **31**, pp. 21–35 (2005).
- [26] Bianchi, S.M., Chakraborty, D., Lucarini, Y., Wagler, A.K.: Location-Domination Type Problems Under the Mycielski Construction. In: *Algorithms and Discrete Applied Mathematics (CALDAM)*. *Lecture Notes in Computer Science* **14508**, pp. 255–269 (2024).
- [27] Bellitto, T., Brosse, C., Lévêque, B., Parreau, A.: Locating-dominating sets in local tournaments. *Discrete Applied Mathematics* **337**, pp. 14–24 (2023).
- [28] Bliznets, I., Hecher, M.: Tight double exponential lower bounds. In Xujin Chen and Bo Li, editors, *Theory and Applications of Models of Computation*, 2024, Hong Kong, China. *Lecture Notes in Computer Science*, Springer **14637**, pp. 124–136 (2024).
- [29] Bondy, J. A.: Induced Subsets. *Journal of Combinatorial Theory* **12**(B), pp. 201–202 (1972).
- [30] Bontridder, K.M.J. de, Halldórsson, B.V., Halldórsson, M.M., Hurkens, C.A.J., Lenstra, J.K., Ravi, R., Stougie, L.: Approximation algorithms for the test cover problem. *Mathematical Programming Series B* **98**, pp. 477–491 (2003).
- [31] Bousquet, N., Chuet, Q., Ravry, V.F., Jacques, A., Morelle, L.: A note on locating sets in twin-free graphs. *arXiv:2405.18162*, (2024).
- [32] Bousquet, N., Deschamps, Q., Lehtilä, T., Parreau, A.: Locating-dominating sets: from graphs to oriented graphs. *Discrete Mathematics* **346**(1), pp. 113124 (2023).
- [33] Brigham, R.C., Carrington, J.R., Vitray, R.P.: Connected graphs with maximum total domination number. *Journal Combinatorial Mathematics and Combinatorial Computing* **34**, pp. 81–96 (2000).

- [34] Brigham, R.C., Chartrand, G., Dutton, R.D., Zhang, P.: Resolving domination in graphs. *Mathematica Bohemica* **128**, pp. 25-36 (2003).
- [35] Cáceres, J., Hernando, C., Mora, M., Pelayo, I.M., Puertas, M.L.: Locating dominating codes: bounds and extremal cardinalities. *Applied Mathematics and Computation* **220**, pp. 38-45 (2013).
- [36] Cappelle, M.R., Gomes, G.C.M, Santos, V.F.: Parameterized algorithms for locating-dominating sets. In Carlos E. Ferreira, Orlando Lee, and Flávio Keidi Miyazawa, editors, *Latin and American Algorithms, Graphs and Optimization Symposium, LAGOS 2021*, *Procedia Computer Science*, Elsevier **195**, pp. 68-76 (2021).
- [37] Cappelle, M.R., Gomes, G.C.M, Santos, V.F.: Parameterized algorithms for locating-dominating sets. *arXiv:2011.14849 [cs.DS]* (2023).
- [38] Chang, H.Y., Manning, E., Metze G.: *Fault Diagnosis of Digital Systems*, John Wiley, New York, (1970).
- [39] Chakraborty, D., Foucaud, F., Hakanen, A., Henning, M.A., Wagler, A.K.: Progress towards the two-thirds conjecture on locating-total dominating sets. *Discrete mathematics* **347**, 114176 (2024).
- [40] Chakraborty, D., Foucaud, F., Henning, M.A.: Identifying open codes in trees and 4-cycle-free graphs of given maximum degree. *arXiv:2407.09692 [math.CO]* (2024).
- [41] Chakraborty, D., Foucaud, F., Henning, M.A., Lehtilä, T.: Identifying codes in graphs of given maximum degree: Characterizing trees. *arxiv:2403.13172 [math.CO]* (2024).
- [42] Chakraborty, D., Foucaud, F., Henning, M.A., Lehtilä, T.: Identifying codes in triangle-free graphs of bounded maximum degree. *arXiv preprint arXiv:2403.17877 [math.CO]* (2024).
- [43] Chakraborty, D., Foucaud, F., Lehtilä, T.: Identifying codes in bipartite graphs of given maximum degree. *Proceedings of LAGOS 2023*, *Procedia Computer Science* **223**, pp. 157-165 (2023).
- [44] Chakraborty, D., Foucaud, F., Majumdar, D., Tale, P.: Tight (Double) Exponential Bounds for Identification Problems: Locating-Dominating Set and Test Cover. *International Symposium on Algorithms and Computation (ISAAC 2024) Leibniz International Proceedings in Informatics* **322**, pp. 19:1-19:18 (2024).
- [45] Chakraborty, D., Foucaud, F., Majumdar, D., Tale, P.: Structural parameterization of Locating-Dominating Set and Test Cover. *arXiv:2411.17948 [cs.DS]* (to appear in the 14th International Conference on Algorithms and Complexity (CIAC, June, 2025, Rome, Italy) (2024).
- [46] Chakraborty, D., Foucaud, F., Nandi, S., Sen, S., Supraja, D.K.: On locating and neighbor-locating colorings of sparse graphs. *Discrete Applied Mathematics* **358**, pp. 366-381 (2024).
- [47] Chakraborty, D., Foucaud, F., Parreau, A., Wagler, K.A.: On three domination-based identification problems in block graphs. *Fundamenta Informaticae* **191**(3-4), pp. 197-229 (2024).
- [48] Chakraborty, D., Hakanen, A., Lehtilä, T.: The  $n/2$ -bound for locating-dominating sets in subcubic graphs. *arXiv:2406.19278 [math.CO]* (2024).
- [49] Chakraborty, D., Nandi, S., Sen, S., Supraja, D.K.: A linear algorithm for radio  $k$ -coloring of powers of paths having small diameters. *Journal of Computer and System Sciences* **147**, 103577 (2024).

- [50] Chakraborty, D., Sandeep, R.B.: Contracting Edges to Destroy a Pattern: A Complexity Study. In: Fernau, H., Jansen, K., editors, *Fundamentals of Computation Theory (FCT) 2023*. Lecture Notes in Computer Science, Springer **14292**, pp. 118-131 (2023).
- [51] Chakraborty, D., Wagler, A.K.: On lower bounds for the size of several separating-dominating codes in graphs. *arXiv:2412.17469 [math.CO]* (2024).
- [52] Chakraborty, D., Wagler, A.K.: Open-Separating Dominating Codes in Graphs. *Combinatorial Optimization (ISCO 2024)* Lecture Notes in Computer Science **14594**, pp. 137-151 (2024).
- [53] Chakraborty, D., Wagler, A.K.: On full-separating sets in graphs. *Algorithms and Discrete Applied Mathematics (CALDAM 2025)* Lecture Notes in Computer Science **15536**, pp. 73-84 (2025).
- [54] Chakraborty, D., Wagler, A.K.: On full-separating sets in graphs. *arXiv:2407.10595 [math.CO]* (2024).
- [55] Chalopin, J., Chepoi, V., Mc Inerney, F., Ratel, S.: Non-clashing teaching maps for balls in graphs. *arXiv:2309.02876 [cs.CC]*, (2023).
- [56] Chandran, L.S., Issac, D., Karrenbauer, A.: On the parameterized complexity of biclique cover and partition. In J. Guo and D. Hermelin, editors, *11th International Symposium on Parameterized and Exact Computation, IPEC 2016, LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik* **63**, pp. 1-13 (2016).
- [57] Charbit, E., Charon, I., Cohen, G.D., Hudry, O., Lobstein, A.: Discriminating codes in bipartite graphs: bounds, extremal cardinalities, complexity. *Advances in Mathematics of Communication* **2**(4), pp. 403-420 (2008).
- [58] Charon, I., Hudry, O., Lobstein, A.: Minimizing the size of an identifying or locating-dominating code in a graph is np-hard. *Theoretical Computer Science* **290**(3), pp. 2109-2120 (2003).
- [59] Charon, I., Hudry, O., Lobstein, A.: Extremal cardinalities for identifying and locating-dominating codes in graphs. *Discrete Mathematics* **307**(3-5), pp. 356-366 (2007).
- [60] Charon, I., Hudry, O., Lobstein, A.: Locating-domination and identification. *Topics in Domination in Graphs*, pp. 251-299 (2020).
- [61] Chlebus, B.S., Nguyen, S.H.: On finding optimal discretizations for two attributes. In *Proceedings of the First International Conference on Rough Sets and Current Trends in Computing* **1424**, pp. 537-544 (1998).
- [62] Cho, E.K., Choi, I., Park, B.: On independent domination of regular graphs. *Journal of Graph Theory* **103**, pp. 159-170 (2023).
- [63] Cho, E.K., Kim, J., Kim, M., Oum, A.: Independent domination of graphs with bounded maximum degree. *Journal of Combinatorial Theory, Series B* **158**, pp. 341-352 (2023).
- [64] Chvátal, V.: Mastermind. *Combinatorica* **3**(3), pp. 325-329 (1983).
- [65] Claverol, M., García, A., Hernández, G., Hernando, C., Maureso, M., Mora, M., Tejel, J.: Metric dimension of maximal outerplanar graphs. *Bulletin of the Malaysian Mathematical Sciences Society* **44**, pp. 2603-2630 (2021).
- [66] Cobham, C.: The intrinsic computational difficulty of functions. *Proceedings of the second International Congress for Logic. Methodology and Philosophy of Science, Jerusalem*, (1964).
- [67] Cockayne, E.J., Dawes, R.M., Hedetniemi, S. T.: Total domination in graphs. *Networks* **10**, pp. 211-219 (1980).

- [68] Colbourn C. J., Slater P. J., Stewart L. K.: Locating dominating sets in series parallel networks. *Congressus Numerantium* **56**, pp. 135-162 (1987).
- [69] Cook, S.: The complexity of theorem proving procedures. *Proceedings of the Third Annual ACM Symposium on Theory of Computing, STOC '71*, pp. 151-158 (1971).
- [70] Coolsaet, K., D'hondt, S., Goedgebeur, J.: House of Graphs 2.0: A database of interesting graphs and more. *Discrete Applied Mathematics* **325**, pp. 97-107 (2023). <https://houseofgraphs.org>
- [71] Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: *Introduction to Algorithms*, Third Edition. The MIT Press (2009).
- [72] Cornuéjols, G.: *Combinatorial Optimization: Packing and Covering*, SIAM, CBMS **74**, (2001).
- [73] Coudert, D., Ducoffe, G., Popa, A.: Fully polynomial FPT algorithms for some classes of bounded clique-width graphs. *ACM Transactions on Algorithms* **15**(3), pp. 1-57 (2019).
- [74] Courcelle, B.: The monadic second-order logic of graphs. I. recognizable sets of finite graphs. *Information and computation* **85**(1), pp. 12-75 (1990).
- [75] Courcelle, B., Makowsky, J.A., Rotics, U.: Linear time solvable optimization problems on graphs of bounded clique-width. *Theory of Computer Systems* **33**(2), pp. 125-150 (2000).
- [76] Cygan, M., Fomin, F.V., Kowalik, L., Lokshtanov, D., Marx, D., Pilipczuk, M., Pilipczuk, M., Saurabh, S.: *Parameterized Algorithms*. Springer, (2015).
- [77] Cygan, M., Pilipczuk, M., Pilipczuk, M.: Known algorithms for edge clique cover are probably optimal. *SIAM Journal on Computing* **45**(1), pp. 67-83 (2016).
- [78] Devijver, P.A., Kittler, J.: *Pattern Recognition: A Statistical Approach*, Prentice-Hall, Englewood Cliffs, NJ, (1982).
- [79] Diestel, R.: *Graph Theory*, 4th Edition. Graduate texts in mathematics, Springer **173** (2012).
- [80] Dorbec, P., Henning, M.A., Montassier, M., Southey, J.: Independent domination in cubic graphs. *Journal of Graph Theory* **80**, pp. 329-349 (2015).
- [81] Downey, R.G., Fellows, M.R.: Fixed-parameter tractability and completeness I: basic theory. *SIAM Journal on Computing* **24**, pp. 873-921 (1995).
- [82] Downey, R.G., Fellows, M.R.: *Parameterized Complexity*. Springer New York, NY, (1999).
- [83] Edmonds, J.: Paths, trees, and flowers. *Canadian Journal of Mathematics* **17**, pp. 449-467 (1965).
- [84] Erdős, P.: Some combinatorial, geometric and set theoretic problems in measure theory. In: Kölzow, D., Maharam-Stone, D. (eds) *Measure Theory Oberwolfach 1983. Lecture Notes in Mathematics*, Springer, Berlin, Heidelberg **1089**, pp. 321-327 (1984).
- [85] Focke, J., Frei, F., Li, S., Marx, D., Schepper, P., Sharma, R., Wegrzycki, K.: Hitting meets packing: How hard can it be? *arXiv:2402.14927 [cs.DS]* (2024).
- [86] Fomin, F.V., Lokshtanov, D., Saurabh, S., Zehavi, M.: *Kernelization: theory of parameterized preprocessing*. Cambridge University Press, (2019).
- [87] Foucaud, F.: Combinatorial and algorithmic aspects of identifying codes in graphs. *Data Structures and Algorithms [cs.DS]*. Université Sciences et Technologies - Bordeaux I, (2012). <https://theses.hal.science/tel-00766138>
- [88] Foucaud, F.: Problems related to a conjecture on location-domination in twin-free graphs. *Indian Journal of Discrete Mathematics* **8**(1), pp. 39-41 (2022).



- [89] Foucaud, F., Ghareghani, N., Roshany-Tabrizi, A., Sharifani, P.: Characterizing extremal graphs for open neighbourhood location-domination. *Discrete Applied Mathematics* **302**, pp. 76-79 (2021).
- [90] Foucaud, F., Guerrini, E., M.A., Kovše, M., Naserasr, R., Parreau, A., Valicov, P.: Extremal graphs for the identifying code problem. *European Journal of Combinatorics* **32**, pp. 628-638 (2011).
- [91] Foucaud, F., Gravier, S., Naserasr, R., Parreau, A., Valicov, P.: Identifying codes in line graphs. *Journal of Graph Theory* **73**(4), pp. 425-448 (2013).
- [92] Foucaud, F.: Decision and approximation complexity for identifying codes and locating-dominating sets in restricted graph classes. *Journal of discrete algorithms* **31**, pp. 48-68 (2015).
- [93] Foucaud, F., Galby, E., Khazaliya, L., Li, S., Mc Inerney, F., Sharma, R., Tale, T.: Tight (double) exponential bounds for np-complete problems: Treewidth and vertex cover parameterizations, arXiv:2307.08149 [cs.CC] (2023).
- [94] Foucaud, F., Heydarshahi, S., Parreau, A.: Domination and location in twin-free digraphs. *Discrete Applied Mathematics* **284**, pp. 42-52 (2020).
- [95] Foucaud, F., Henning, M.A.: Location-domination and matching in cubic graphs. *Discrete Mathematics* **339**(4), pp. 1221-1231 (2016).
- [96] Foucaud, F., Henning, M.A.: Locating-total dominating sets in twin-free graphs: a conjecture. *The Electronic Journal of Combinatorics* **23**(3), P3.9 (2016).
- [97] Foucaud, F., Henning, M.A.: Location-domination in line graphs. *Discrete Mathematics* **340**, pp. 3140-3153 (2017).
- [98] Foucaud, F., Henning, M.A., Löwenstein C., Sasse T.: Locating-dominating sets in twin-free graphs. *Discrete Applied Mathematics* **200**, pp. 52-58 (2016).
- [99] Foucaud, F., Klasing, R., Kosowski, A., Raspaud, A.: On the size of identifying codes in triangle-free graphs. *Discrete Applied Mathematics* **160**, pp. 1532-1546 (2012).
- [100] Foucaud, F., Lehtilä, T.: Revisiting and improving upper bounds for identifying codes. *SIAM Journal on Discrete Mathematics* **36**(4), pp. 2619-2634 (2022).
- [101] Foucaud, F., Lehtilä, T.: Bounds and extremal graphs for total dominating identifying codes. *The Electronic Journal of Combinatorics* **30**(3), P3.15 (2023).
- [102] Foucaud, F., Mertzios G., Naserasr R., Parreau A., Valicov P.: Identification, location-domination and metric dimension on interval and permutation graphs. I. Bounds. *Theoretical Computer Science* **668**, pp. 43-58 (2017).
- [103] Foucaud, F., Mertzios G., Naserasr R., Parreau A., Valicov P.: Identification, location-domination and metric dimension on interval and permutation graphs. II. Algorithms and complexity. *Algorithmica* **78**(3), pp. 914-944 (2017).
- [104] Foucaud, F., Perarnau, G.: Bounds for identifying codes in terms of degree parameters. *The Electronic Journal of Combinatorics* **19**, P32 (2012).
- [105] Foucaud, F., Slater, P.J: Private communication, (2013).
- [106] Fichte, J.K., Hecher, M., Morak, M., Thier, P., Woltran, S.: Solving projected model counting by utilizing treewidth and its limits. *Artificial Intelligence* **314**, 103810 (2023).
- [107] Fichte, J.K., Hecher, M., Morak, M., Woltran, S.: Exploiting treewidth for projected model counting and its limits. In *Theory and Applications of Satisfiability Testing 2018, Lecture Notes in Computer Science*, Springer **10929**, pp. 165-184 (2018).

- [108] Ganian, R.: Twin-cover: Beyond vertex cover in parameterized algorithmics. In Dániel Marx and Peter Rossmanith, editors, *Parameterized and Exact Computation - 6th International Symposium, IPEC 2011, Revised Selected Papers, Lecture Notes in Computer Science*, Springer **7112**, pp. 259–271 (2011).
- [109] Garey, M.R., Johnson, D.S.: *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, USA (1990).
- [110] Gravier, S., Moncel, J.: On graphs having a  $V \setminus x$  set as an identifying code. *Discrete Mathematics* **307**(3-5), pp. 432-434 (2007).
- [111] Gravier, S., Moncel, J., Semri, A.: Identifying codes of cycles. *European Journal of Combinatorics* **27**(5), pp. 767-776 (2006).
- [112] Garijo, D., González, A., Márquez A.: The difference between the metric dimension and the determining number of a graph. *Applied Mathematics and Computation* **249**(1), pp. 487–501 (2014).
- [113] Gimbel, J., Gordon, B.D., Nicolescu, M., Umstead, C., Vaiana, N.: Location with dominating sets. *Congressus Numerantium*, pp. 129-144 (2001).
- [114] Goddard, W., Henning, M.A., Lyle, J., Southey, J.: On the independent domination number of regular graphs. *Annals of Combinatorics* **16**, pp. 719-732 (2012).
- [115] Goddard, W., Wash, K.: ID codes in Cartesian products of cliques. *Journal of Combinatorial Mathematics and Combinatorial Computing* **85**, pp. 97-106 (2013).
- [116] Gravier, S., Klasing, R., Moncel J.: Hardness results and approximation algorithms for identifying codes and locating-dominating codes in graphs. *Algorithmic Operations Research* **3**(1), pp. 43–50 (2008).
- [117] Gravier, S., Kovše, M., Mollard, M., Moncel, J., Parreau, A.: New results on variants of covering codes in Sierpiński graphs. *Designs, Codes and Cryptography* **69**(2), pp. 181-188 (2013).
- [118] Habib, M., Paul, C., Viennot, L.: A synthesis on partition refinement: A useful routine for strings, graphs, boolean matrices and automata. In *Proceedings of the 15th Annual Symposium on Theoretical Aspects of Computer Science (STACS 1998)*, Springer-Verlag, pp. 25-38 (1998).
- [119] Hanaka, T., Köhler, N., Lampis, M.: Core stability in additively separable hedonic games of low treewidth, *arXiv:2402.10815 [cs.DS]* (2024).
- [120] Harary, F.: A Characterization of Block-Graphs. *Canadian Mathematical Bulletin* **6**(1), pp. 1-6 (1963).
- [121] Harary F., Melter, R. A.: The metric basis of a graph. *Ars Combinatoria* **2**, pp. 191-195 (1976).
- [122] Harris, D., Narayanaswamy, N.S.: A faster algorithm for vertex cover parameterized by solution size. In Olaf Beyersdorff, Mamadou Moustapha Kanté, Orna Kupferman, and Daniel Lokshtanov, editors, *In 41st International Symposium on Theoretical Aspects of Computer Science, STACS 2024, LIPIcs* **289**, pp. 1-18 (2024).
- [123] Haynes, T.W., Henning, M.A., Howard, J.: Locating and total dominating sets in trees. *Discrete Applied Mathematics* **154**(8), p. 1293-1300 (2006).
- [124] Haynes, T.W., Hedetniemi, S.T., Henning, M.A.: *Structures of Domination in Graphs*, volume 66 of *Developments in Mathematics*. Springer, Cham, (2021).
- [125] Haynes, T.W., Hedetniemi, S.T., Henning, M.A.: *Domination in Graphs: Core Concepts*. Springer Monographs in Mathematics, Springer, Cham, (2023).

- [126] Haynes, T.W., Knisley, D.J., Seier, E., Zou, Y.: A quantitative analysis of secondary RNA structure using domination based parameters on trees. *BMC Bioinformatics* **7**(108), (2006).
- [127] Henning, M.A., Löwenstein, C.: Locating-total domination in claw-free cubic graphs. *Discrete Mathematics* **312**(21), pp. 3107-3116 (2012).
- [128] Henning, M.A., Rad, J.R.: Locating-total domination in graphs. *Discrete Applied Mathematics* **160**(13-14), pp. 1986-1993 (2012).
- [129] Henning, M.A., Yeo, A.: *Total domination in graphs*. Springer, (2013).
- [130] Henning, M.A., Yeo, A.: Distinguishing-Transversal in Hypergraphs and Identifying Open Codes in Cubic Graphs. *Graphs and Combinatorics* **30**, pp. 909-932 (2014).
- [131] Hernando, C., Mora, M., Slater, P.J., Wood, D.R.: Fault-tolerant metric dimension of graphs, *Ramanujan Mathematical Society Lecture Notes* **5**, pp. 81–85 (2008).
- [132] Herva, P., Laihonen, T., Lehtilä, T.: Optimal Local Identifying and Local Locating-dominating Codes. *Fundamenta Informaticae* **191**(3-4), pp. 351-378 (2024).
- [133] Honkala, I., Laihonen, T., Ranto, S.: On strongly identifying codes. *Discrete Mathematics* **254**(1-3), pp. 191-205 (2002).
- [134] Howorka, E.: On metric properties of certain clique graphs. *Journal of Combinatorial Theory, Series B* **27**(1), pp. 67-74 (1979).
- [135] Hulme B. L., Shiver A. W., Slater P. J.: FIRE: A Subroutine for Fire Protection. *Network Analysis*. SAND81-1261, Sandia National Laboratories, Albuquerque, NM (1981).
- [136] Hulme B. L., Shiver A. W., Slater P. J.: Computing Minimum Cost Fire Protection. SAND82-0809, Sandia National Laboratories, Albuquerque, NM (1982).
- [137] Hulme B. L., Shiver A. W., Slater P. J.: A Boolean algebraic analysis of fire protection. *Algebraic and combinatorial methods in operations research* **95**, pp. 215–227 (1984).
- [138] Ian, A.: *Combinatorics of Finite Sets*. Oxford University Press, (1987).
- [139] Impagliazzo, R., Paturi, R.: On the complexity of k-sat. *Journal of Computer Systems and Sciences* **62**(2), pp. 367-375 (2001).
- [140] Impagliazzo, R., Paturi, R., Zane, F.: Which problems have strongly exponential complexity? *Journal of Computer and System Sciences* **63**(4), pp. 512–530 (2001).
- [141] Jean, D.C., Lobstein, A.: Watching systems, identifying, locating-dominating and discriminating codes in graphs: a bibliography. <https://dragazo.github.io/bibdom/main.pdf>
- [142] Jean, D.C., Seo, S.: Fault-Tolerant Locating-Dominating sets with Error-correction. *arXiv:2212.08193*, (2022).
- [143] Jean, D.C., Seo, S.: Error-correcting Identifying Codes. *arXiv:2204.11362*, (2022).
- [144] Jean, D.C., Seo S.: Optimal error-detecting open-locating-dominating set on the infinite triangular grid. *Discussiones Mathematicae Graph Theory* **43**(2), pp. 445-455 (2023).
- [145] Junnila, V., Laihonen, T., Lehtilä, T.: On a conjecture regarding identification in Hamming graphs. *The Electronic Journal of Combinatorics*, pp. P2 (2019).
- [146] Junnila, V., Laihonen, T., Lehtilä, T., Puertas, M.L.: On Stronger Types of Locating-dominating Codes. *Discrete Mathematics and Theoretical Computer Science*, **21**(1), (2019).
- [147] Junnila, V., Laihonen, T.: Tolerant location detection in sensor networks. *Advances in Applied Mathematics* **112**, 101938 (2020).

- [148] Karp, R.M.: Reducibility among combinatorial problems. In R. E. Miller and J.W. Thatcher, editors, *Complexity of Computer Computations*, Plenum Press, pp. 85-103 (1972).
- [149] Karpovsky, M.G., Chakrabarty, K., Levitin, L.B.: On a new class of codes for identifying vertices in graphs. *IEEE transactions on information theory* **44**(2), pp. 599–611 (1998).
- [150] Kellerhals, L. Koana, T.: Parameterized complexity of geodetic set. *Journal of Graph Algorithms and Applications* **26**(4), pp. 401-419 (2022).
- [151] Kim, J.H., Pikhurko, O., Spencer, J., Verbitsky, O.: How complex are random graphs in First Order logic? *Random Structures and Algorithms* **26**(1-2), pp. 119-145 (2005).
- [152] Kloks, T.: *Treewidth, Computations and Approximations*. Springer, (1994).
- [153] Korhonen, T.: A single-exponential time 2-approximation algorithm for treewidth. In 2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS 2021), pp. 184–192 (2022).
- [154] Kostochka, A., Stodolsky, B.: An upper bound on the domination number of  $n$ -vertex connected cubic graphs. *Discrete Mathematics* **309**(5), pp. 1142–1162 (2009).
- [155] Kratsch, S., Philip, G., Ray, S.: Point line cover: The easy kernel is essentially tight. *ACM Transactions on Algorithms* **12**(3), pp. 1-16 (2016).
- [156] Laifenfeld, M., Trachtenberg, A., Cohen, R., Starobinski, D.: Joint monitoring and routing in wireless sensor networks using robust identifying codes. *Proceedings of IEEE Broadnets*, pp. 197-206 (2007).
- [157] Lampis, M.: Algorithmic meta-theorems for restrictions of treewidth. *Algorithmica* **64**(1), pp. 19-37 (2012).
- [158] Levin, L.A.: Universal Sequential Search Problems, *Problemy Peredachi Informatsii* **9**(3), pp. 115–116; *Problems of Information Transmission* **9**(3), pp. 265–266 (1973).
- [159] Lokshtanov, D., Marx, D., Saurabh, S.: Known algorithms on graphs of bounded treewidth are probably optimal. *ACM Trans. Algorithms* **14**(2), pp. 1-30 (2018).
- [160] Marx, D. Mitsou, V.: Double-exponential and triple-exponential bounds for choosability problems parameterized by treewidth. In 43rd International Colloquium on Automata, Languages, and Programming 2016, *LIPIcs* **55**, pp. 1-15 (2016).
- [161] Marinos P.N.: Derivation of minimal complete sets of test-input sequences using Boolean differences. *IEEE Transactions on Computers*, **20**, pp. 25-32 (1971).
- [162] Miller, M., Rajan, R.S., Jayagopal, R., Rajasingh, I.: A note on the locating-total domination in graphs. *Discussiones Mathematicae Graph Theory* **37**, pp. 745-754 (2017).
- [163] Moncel, J.: Codes identifiants dans les graphes. PhD thesis, Université Joseph Fourier Grenoble I, France (2005).
- [164] Moncel, J.: On graphs on  $n$  vertices having an identifying code of cardinality  $\log_2(n + 1)$ . *Discrete Applied Mathematics* **154**(14), pp. 2032-2039 (2006).
- [165] Moore, C.; Mertens, S.: *Symmetry-breaking and NAESAT. The Nature of Computation*, Oxford University Press, pp. 133-138 (2011).
- [166] Moret, B.M.E., Shapiro, H.D.: On minimizing a set of tests, *SIAM Journal on Scientific and Statistical Computing*, **6**(4), pp. 983–1003 (1985).
- [167] Müller, T., Sereni, J.S.: Identifying and locating-dominating codes in (random) geometric networks. *Combinatorics, Probability and Computing* **18**(6) pp. 925-952 (2009).

- [168] Narendra, P.M., Fukunaga, K.: A branch-and-bound algorithm for feature subset selection. *IEEE Transactions on Computers* **26**, pp. 917-922 (1977).
- [169] Nobile, P., Sassano, A.: Facets and lifting procedures for the set covering polytope. *Mathematical Programming* **45**, pp. 111-137 (1989).
- [170] Ore, O.: *Theory of Graphs*. American Mathematical Society Colloquium Publications (1962).
- [171] Pandey, A.: Open neighborhood locating-dominating set in graphs: Complexity and algorithms. *International Conference on Information Technology (ICIT)*, Bhubaneswar, India, pp. 1-6 (2015).
- [172] Panda, B.S., Pandey, A.: Algorithmic aspects of open neighborhood location-domination in graphs. *Discrete Applied Mathematics* **216**, pp. 290-306 (2017).
- [173] Panda, B.S., Goyal, P., Pradhan, D.: Differentiating-total domination: Approximation and hardness results. *Theoretical Computer Science* **876**, pp. 45-58 (2021).
- [174] Payne, R.W., Preece, D.A.: Identification keys and diagnostic tables: a review. *Journal of the Royal Statistical Society: Series A* **143**, pp. 253-292 (1980).
- [175] Pilipczuk, M.: Problems parameterized by treewidth tractable in single exponential time: A logical approach. In *Mathematical Foundations of Computer Science 2011 - MFCS 2011*, *Lecture Notes in Computer Science*, Springer **6907**, pp. 520-531 (2011).
- [176] Poureidi, A., Farshi, M.: Algorithmic results on locating-total domination in graphs. *Discrete Applied Mathematics* **334**, pp. 36-44 (2023).
- [177] Rahbani, H., Rad, N.J., Sadeghi, M.R.: A note on the complexity of locating-total domination in graphs. *Theoretical Computer Science* **799**, pp. 32-39 (2019).
- [178] Rall, D.F., Wash, K.: Identifying codes of the direct product of two cliques. *European Journal of Combinatorics* **36**, pp. 159-171 (2014).
- [179] Rao, N.S.V.: Computational complexity issues in operative diagnosis of graph-based systems. *IEEE Transactions on Computers*, **42**(4), pp. 447-457 (1993).
- [180] Ray, S., Ungrangsi, R., Pellegrini, F. de, Trachtenberg, A., Starobinski, D.: Robust location detection in emergency sensor networks. *IEEE Journal on Selected Areas in Communications* **22**(6), pp. 1016-1025 (2004).
- [181] Reed, B.: Paths, stars and the number three. *Combinatorics, Probability and Computing* **5**(3), pp. 277-295 (1996).
- [182] Rooij, J.M.M. van, Bodlaender, H.L., Rossmanith, P.: Dynamic programming on tree decompositions using generalised fast subset convolution. In *Amos Fiat and Peter Sanders, editors, Algorithms - ESA 2009, 17th Annual European Symposium, Copenhagen, Denmark, Lecture Notes in Computer Science* **5757**, pp. 566-577 (2009).
- [183] Sau, I., Souza, U.S.: Hitting forbidden induced subgraphs on bounded treewidth graphs. *Information and Computation* **281**, 104812 (2021).
- [184] Slater P. J.: Leaves of tree. *Proceedings of the Sixth S. E. Conference on Combinatorics, Graph Theory and Computing, Utilitas Math*, pp. 549-559 (1975).
- [185] Slater P. J.: Domination and location in acyclic graphs. *Networks*, **17**(1) pp. 55-64 (1987).
- [186] Slater, P. J.: Dominating and reference sets in a graph. *Journal of Mathematical and Physical Sciences* **22**, pp. 445-455 (1988).

- [187] Slater, P. J.: Fault tolerant locating dominating sets. *Discrete Mathematics* **249**, pp. 179–189 (2002).
- [188] Slater, P. J.: A framework for faults in detectors within network monitoring systems. *WSEAS Transactions on Mathematics* **12**(9), pp. 911–916 (2013).
- [189] Sebö, A., Tannier, E.: On metric generators of graphs. *Mathematics of Operations Research* **29**(2), pp. 383–393 (2004).
- [190] Seo, S.J., Slater, P.J.: Open neighborhood locating dominating sets. *Australasian Journal of Combinatorics* **46**, pp. 109–120 (2010).
- [191] Seo, S.J., Slater, P.J.: Open neighborhood locating-dominating in trees. *Discrete Applied Mathematics* **159**, pp. 484–489 (2011).
- [192] Seo, S.J., Slater, P.J.: Fault tolerant detectors for distinguishing sets in graphs. *Discussiones Mathematicae Graph Theory* **35**, pp. 797–818 (2015).
- [193] Seo, S., Slater, P.J.: Generalized set dominating and separating systems. *Journal of Combinatorial Mathematics and Combinatorial Computing* **104**, pp. 15–29 (2018).
- [194] Sewell, J.L.: Vertex-based distinguishing collections. *Dissertations* **134** (2017).
- [195] Skaggs, R.D.: Identifying vertices in graphs and digraphs. PhD thesis, (2007).
- [196] Thomassé, S., Yeo, A.: Total domination of graphs and small transversals of hypergraphs. *Combinatorica* **27**, pp. 473–487 (2007).
- [197] Tovey, C.: A simplified np-complete satisfiability problem. *Discrete Applied Mathematics* **8**(1), pp. 85–89 (1984).
- [198] Ungrangsi, R., Trachtenberg, A., Starobinski, D.: An implementation of indoor location detection systems based on identifying codes. In *Proceedings of Intelligence in Communication Systems, INTELLCOMM, Lecture Notes in Computer Science* **3283**, pp. 175–189 (2004).
- [199] West, D.B.: *Introduction to Graph Theory* (2nd Edition). Prentice Hall, (2001).
- [200] Willcox, W. R., Lapage, S. P.: Automatic construction of diagnostic tables. *The Computer Journal* **15**, pp. 263–267 (1972).
- [201] Willcox, W. R., Lapage, S. P., Holmes, I.: A review of numerical methods in bacterial identification. *Antonie van Leeuwenhoek*, **46** pp. 233–299 (1980).



# List of Tables

2.1	Known bounds for LD-numbers in some common graph families . . . . .	52
2.2	Graph families for which the conjectured $n$ -half upper bound in [112] holds . . . . .	53
2.3	Known bounds for LTD-numbers of some common graph families . . . . .	54
2.4	Graph families for which the conjectured $n$ -two-thirds upper bound in [96] holds . .	54
2.5	Known bounds for ID-numbers of some common graph families . . . . .	56
2.6	Known bounds for ITD-numbers of some common graph families . . . . .	56
2.7	Known bounds for OTD-numbers of some common graph families . . . . .	57
2.8	Known bounds for the OD-numbers of some common graph families . . . . .	57
2.9	Known bounds for FD-numbers of some common graph families . . . . .	58
2.10	Known bounds for FTD-numbers of some common graph families . . . . .	58
2.11	Complexity classes to which LD-CODE belongs . . . . .	59
2.12	Complexity classes to which LTD-CODE belongs . . . . .	59
2.13	Complexity classes to which ID-CODE belongs . . . . .	60
2.14	Complexity classes to which ITD-CODE belongs . . . . .	60
2.15	Complexity classes to which OD-CODE belongs . . . . .	60
2.16	Complexity classes to which OTD-CODE belongs . . . . .	61
2.17	Complexity classes to which FD-CODE belongs . . . . .	61
2.18	Complexity classes to which FTD-CODE belongs . . . . .	62
5.1	Trees requiring a positive constant $c$ for Conjecture 5.2 . . . . .	148
5.2	Known triangle-free graphs requiring a positive constant $c$ for Conjecture 5.2 . . . .	170
9.1	Partition of $V(G)$ with respect to proof of Theorem 9.1 . . . . .	256
9.2	The five rooted tree gadgets used in the proof of Theorem 9.6 . . . . .	278
10.1	Classification of codes into their original, local and non-local versions . . . . .	289





# List of Figures

1.1	Schematic representation of a building plan modelled as a graph . . . . .	14
1.2	Example of a graph . . . . .	15
1.3	Example of a (minimum) LD-code of a graph . . . . .	17
1.4	Example of a (minimum) ID-code of a graph . . . . .	19
1.5	Example of a (minimum) OTD-code of a graph . . . . .	20
1.6	Example of a (minimum) FTD-code of a graph . . . . .	21
2.1	Examples of graphs of various categories . . . . .	28
2.2	Examples of dominating and total-dominating sets of graphs . . . . .	29
2.3	Example of a rooted tree . . . . .	32
2.4	Half-graphs and their complements: examples of bipartite and cobipartite graphs . .	32
2.5	Example of a chordal graph . . . . .	33
2.6	Examples of thin and thick headless spiders . . . . .	33
2.7	Example of layers, articulation vertices and non-articulation vertices of a connected block graph. . . . .	34
2.8	Example of interval and unit interval graphs . . . . .	35
2.9	Example of a permutation graph . . . . .	35
2.10	Example of line graph of a 3-star . . . . .	36
2.11	Examples of thin suns . . . . .	36
2.12	Example of a tree decomposition . . . . .	42
2.13	Example of vertex cover number (and a minimum-ordered vertex cover) of a graph .	43
2.14	Example of neighborhood diversity of a graph . . . . .	43
2.15	Example of twin-cover number (and a minimum-ordered twin-cover) of a graph . . .	43
2.16	Example of distance to clique of a graph . . . . .	44
2.17	Example of feedback edge set number (and a minimum-ordered feedback edge set) of a graph . . . . .	44
2.18	Schematic representation of structural parameters of graphs . . . . .	45
2.19	Examples of (open and closed) twins in graphs . . . . .	45
3.1	Example of a hypergraph . . . . .	64
3.2	Schematic representation (directed multicube $C_{\text{CODES}}$ ) of relations between code numbers	68
4.1	Tight example of upper bound for LD-numbers of block graphs . . . . .	93
4.2	Rules (1) to (5) used in Theorem 4.3 to prove the $n$ -half upper bound conjecture for block graphs . . . . .	93
4.3	Tight example of block graph whose LD-number attains the $n$ -half upper bound conjecture . . . . .	96
4.4	Tight example of block graph whose LD-number attains the lower bound in terms of orders of graphs in Theorem 4.4 . . . . .	99
4.5	Example of subcubic graph showing that the $n$ -half upper bound bound does not hold for LTD-numbers . . . . .	102
4.6	Extremal cubic graphs (presented in [95]) whose LD-numbers achieve the $n$ -half upper bound . . . . .	102

4.7	Illustrations for (a) Case 1.1.2, (b) Case 1.2.3 and (c) Case 1.2.3.2 linked with the proof of Proposition 4.4 showing the $n$ -half conjecture for subcubic graphs . . . . .	104
4.8	Illustrations for (a) Case 1.2.3.3, (b) Case 1.3.2 and (c) Case 1.3.3 linked with proof of Proposition 4.4 showing the $n$ -half conjecture for subcubic graphs . . . . .	106
4.9	Illustrations for (a) Case 2.1, (b) Case 2.2 and (c) Case 2.2.2 linked with proof of Proposition 4.4 showing the $n$ -half conjecture for subcubic graphs . . . . .	108
4.10	A list of pairwise non-isomorphic subcubic graphs linked with the proof of Theorem 4.7 showing the $n$ -half conjecture for subcubic graphs with open twins of degree 3 allowed	112
4.11	Example of a subcubic graph containing open twins of degree 1 for which the $n$ -half upper bound conjecture fails . . . . .	119
4.12	Example of a subcubic graph containing open twins of degree 2 for which the $n$ -half upper bound conjecture fails . . . . .	119
4.13	Example of $r$ -regular graph with $r \geq 4$ for which the $n$ -half upper bound conjecture fails . . . . .	120
4.14	Tight example of a subcubic graph for the $n$ -half upper bound conjecture . . . . .	120
4.15	Tight example of a split graph for the $n$ -two-thirds upper bound conjecture . . . . .	124
4.16	Example of a block graph and its auxiliary tree construction . . . . .	125
4.17	Twin-free block graph $G$ . The dotted boxes mark the blocks $X$ , $X''$ and $Y$ of $G$ as in the proof of Theorem 4.11. . . . .	128
4.18	Tight example of a block graph for the $n$ -two-thirds upper bound conjecture . . . . .	130
4.19	Tight example of a subcubic graph for the $n$ -two-thirds upper bound conjecture . . . . .	135
4.20	Example of a cubic graph $G$ on $n = 10$ vertices containing both open and closed twins for which $\gamma^{\text{LD}}(G) = \frac{n}{2}$ . . . . .	136
5.1	Extremal block graph whose ID-number attains the lower bounds in terms of orders of graphs in Theorem 5.2 . . . . .	144
5.2	The family $\mathcal{T}_3$ of 12 trees of maximum degree 3 requiring $c > 0$ in Conjecture 5.2 . . . . .	149
5.3	Examples of paths appended with stars . . . . .	156
5.4	Appended stars and paths illustrating the cases in Lemma 5.11 . . . . .	157
5.5	Tree $T_7 \in \mathcal{T}_3$ : $\text{diam}(T_7) = 6$ and $\gamma^{\text{ID}}(T_7) = 9$ . . . . .	161
5.6	Graph $Z$ : $\gamma^{\text{ID}}(Z) = 8$ . . . . .	162
5.7	Tree $T_{t,\Delta}$ depicting the asymptotically tight graphs described in Proposition 5.4 . . . . .	165
5.8	Figure depicting the general setting of the proof of Theorem 5.8 . . . . .	171
5.9	Figure depicting the setting of the proof of Theorem 5.8 after applying Claim 2 . . . . .	179
5.10	Figure depicting the setting of Claim 4 in the proof of Theorem 5.8 . . . . .	182
5.11	Construction of the set $B^*$ in the proof of Claim 4 in the proof of Theorem 5.8 . . . . .	183
6.1	The bull graph . . . . .	194
6.2	The graphs $K_4 \triangleright_q P_4$ and $K_4 \triangleright_q Q_5$ occurring the Theorem 6.1 for the upper bound of OTD-numbers of block graphs . . . . .	195
6.3	Tight examples of a block graph whose OTD-numbers attain the upper bound in Theorem 6.1 . . . . .	197
6.4	Extremal block graphs whose OTD-numbers attain the lower bounds in Theorem 6.2 . . . . .	199
6.5	Example of a minimum OTD-code of an $n$ -cycle with $n \geq 3$ and $n \neq 4$ . . . . .	204
6.6	The subdivided and the reduced subdivided star . . . . .	204
6.7	The canonical sets of a general tree and the trees $T_2$ and $T_3^*$ in $\mathcal{T}$ linked with the proof of Theorem 6.6 . . . . .	207
6.8	Possible trees occurring in the proof of Claim 3 of Theorem 6.6 . . . . .	209
6.9	A possible tree occurring in the proof of Claim 3 of Theorem 6.6 . . . . .	210
6.10	Two possible subtrees in $T$ occurring in the proof of Theorem 6.6 . . . . .	212
6.11	Possible graphs $G \cong T_k + e$ as in Proposition 6.5 . . . . .	213
6.12	A possible structure of the graph $G$ in the proof of Theorem 6.5 . . . . .	214
6.13	Tight example of a tree attaining the upper bound of Theorem 6.5 . . . . .	215
6.14	A subcubic graph $G$ of order $n$ satisfying $\gamma^{\text{OTD}}(T) = \frac{5}{6}n$ . . . . .	215

6.15	Examples of OD-codes of half-graphs and subdivided stars . . . . .	220
7.1	Examples of minimum FTD-codes of paths $P_n$ with $n \geq 4$ and cycles $C_n$ with $n \geq 5$	231
7.2	Examples of FD- and FTD-codes in half-graphs and headless spiders . . . . .	233
8.1	Reduction 8.1 from LSAT to OD-CODE and $OD \neq OTD$ . . . . .	242
8.2	Reduction 8.2 from 3-SAT to FD-CODE, FTD-CODE and $FD \neq FTD$ . . . . .	247
9.1	Example of the graph constructed by Reduction 9.1 to prove a $2^{o(k^2)} \cdot n^{\mathcal{O}(1)}$ running-time lower bound of LD-CODE when parameterized by solution size $k$ . . . . .	254
9.2	Example of the graph constructed by Reduction 9.2 to prove no- $\mathcal{O}(n^{2-\epsilon})$ order incompressibility of LD-CODE . . . . .	258
9.3	Example of the graph constructed by Reduction 9.3 to prove the $2^{2^{o(\text{tw})}} \cdot \text{poly}(n)$ running time lower bounds of LD-CODE when parameterized by treewidth $\text{tw}$ . . . .	267
9.4	Illustration of the dynamic programming showing the $2^{\mathcal{O}(\text{vc} \log \text{vc})} \cdot n^{\mathcal{O}(1)}$ running time of LD-CODE when parameterized by vertex cover number $\text{vc}$ . . . . .	270
9.5	Instance of the ANNOTATED RED-BLUE PARTITION REFINEMENT problem linked with LD-CODE parameterized by vertex cover number . . . . .	272