The complexity of partially dominating graphs

David C. Kutner[1]
Department of Computer Science, Durham University, UK

Iain A. Stewart
Department of Computer Science, Durham University, UK

Abstract

What is the complexity of dominating part of a graph? We formalize this question as PARTIAL DOMINATION: given a graph H and a subset T of its vertices, determine its domination number $\gamma(H[T])$. We study the complexity landscape of this problem when the "host graph" H is restricted to some graph class $\mathcal{G}$. PARTIAL DOMINATION restricted to $\mathcal{G}$ is at least as hard as DOMINATING SET is restricted the same, and at most as hard as DOMINATING SET is restricted to the class of induced subgraphs of members of $\mathcal{G}$ (its hereditary closure). Assuming the Exponential Time Hypothesis, this is tight in that we are able to exhibit graph classes with every computational hardness combination the proposition allows. For example, we construct a class $\mathcal{F}$ such that: DOMINATING SET is in P restricted to $\mathcal{F}$, PARTIAL DOMINATION is NP-intermediate restricted to $\mathcal{F}$, and DOMINATING SET is NP-complete restricted to the hereditary closure of $\mathcal{F}$.

2012 ACM Subject Classification Mathematics of computing → Graph theory

Keywords and phrases computational complexity graph classes hypercubes graph domination.

[Paper under preparation - draft from February 27, 2025]

1 Introduction

Our questions stem from a recently published [ref pending DOI] journal paper, an extended abstract for which was presented at ALGOWIN 2024 [2]. In that work, it was theoretically interesting and practically relevant to determine the computational complexity of DOMINATING SET restricted to induced subgraphs of hypercubes, but with the subtlety that we required the full hypercube (i.e. the induced supergraph) to also be in the input. It was possible, but nontrivial, to prove NP-completeness of this case. This raised the question: is there a class of graphs such that including the supergraph as a witness makes finding a dominating set strictly easier? (In the formalism introduced below: is there some $\mathcal{G}$ such that PARTIAL DOMINATION($\mathcal{G}$) is strictly easier than DOMINATING SET($\mathcal{G}_I$)?)

2 Prerequisites and definitions

Throughout this work, we assume P≠NP (or our questions become meaningless) and, more strongly, the Exponential Time Hypothesis (ETH) - though we expect this second requirement may be unnecessary to obtain our results.

We begin with some basic definitions from graph theory. Let $G = (V, E)$ be any simple undirected graph. We define the *open neighborhood* of a vertex v to be $N(v) := \{u : (u, v) \in E\}$, and its *closed neighborhood* $N[v] := N(v) \cup \{v\}$. Likewise, for any set of vertices S, we define $N[S] := \cup_{v \in S} N[v]$ and $N(S) := N[V] \setminus S$. A *dominating set* is a set of vertices $S \subseteq V$ such that $N[v] = V$. The *domination number* $\gamma(G)$ is the least number of vertices in any dominating set of G. The decision problem DOMINATING SET takes as input a graph G and integer k and asks whether $\gamma(G_{\leq}k)$. Where $S \subseteq V$ is a set of vertices in the graph, we

[1]Corresponding author.

denote $G[S]$ the subgraph of G induced by S. That is, $G[S]$ has S as its set of vertices and as edges exactly those edges of G with both endpoints incident to a vertex in S. We shall also make use of the following:

▶ **Definition 1** (Induced subgraph/supergraph, hereditary closure). *Where $T \subseteq V$ is a set of vertices in the graph, we denote $G[T]$ the subgraph of G induced by T. That is, $G[T]$ has T as its set of vertices and as edges exactly those edges of G with both endpoints incident to a vertex in T. We say G is an* induced subgraph *of H, and that H is an* induced supergraph *of G, if there exists some T such that $G = H[T]$.*

Let $\mathcal{G}$ be a class of graphs (the object $\mathcal{G}$ is formally an infinite set of graphs). Then we denote $\mathcal{G}_I$ the family $\{H : H$ is an induced subgraph of some graph $G \in \mathcal{G}\}$. We say $\mathcal{G}_I$ is the hereditary closure *of $\mathcal{G}$. If $\mathcal{G} = \mathcal{G}_I$, then we say the $\mathcal{G}$ is* hereditary*.*

▶ **Definition 2** ($\Pi(\mathcal{G})$). *Let Π be a graph problem with inputs $G, X_1, \ldots, X_\ell$, with G an undirected graph (possibly with $\ell = 0$, in which case Π takes just a graph as input). We denote $\Pi(\mathcal{G})$ the restriction of Π to the graph class $\mathcal{G}$, that is, $\Pi(\mathcal{G})$ has as instances exactly those instances of Π satisfying $G \in \mathcal{G}$.*

We introduce a new graph problem, which is subtly different from the restriction of Dominating Set, as we shall see.

> Partial Domination
> *Input:* Simple undirected graph $G = (V, E)$; set $T \subseteq V$; integer k.
> *Question:* Is $(G[T], k)$ a yes-instance of Dominating Set? Equivalently, is there some set $S \subseteq T$ with $|S| \leq k$ such that $T \subseteq N[S]$?

3 Known and immediate results

It is obvious from our definitions above that, for any class of graphs $\mathcal{G}$, if Dominating Set($\mathcal{G}$) is NP-complete, then Partial Domination($\mathcal{G}$) is NP-complete. Further, it is natural and immediate that, if Partial Domination($\mathcal{G}$) is NP-complete, then Dominating Set($\mathcal{G}_I$) is NP-complete. Polynomial solvability results propagate in the opposite direction. As implications, we may write:

$$\forall \mathcal{G} : \text{Dominating Set}(\mathcal{G}) \text{ is NP-c} \implies \text{Partial Domination}(\mathcal{G}) \text{ is NP-c} \tag{1}$$

$$\forall \mathcal{G} : \text{Partial Domination}(\mathcal{G}) \text{ is NP-c} \implies \text{Dominating Set}(\mathcal{G}_I) \text{ is NP-c} \tag{2}$$

$$\forall \mathcal{G} : \text{Dominating Set}(\mathcal{G}) \text{ is in P} \impliedby \text{Partial Domination}(\mathcal{G}) \text{ is in P} \tag{3}$$

$$\forall \mathcal{G} : \text{Partial Domination}(\mathcal{G}) \text{ is in P} \impliedby \text{Dominating Set}(\mathcal{G}_I) \text{ is in P} \tag{4}$$

On the other hand, it is not immediately clear whether the reverse implications hold. We shall exhibit classes of graphs which prove they do not.

4 Contribution

We define, in Table 1, *families of classes of graphs* $\mathfrak{A}, \ldots, \mathfrak{J}$. Every graph class $\mathcal{G}$ belongs to exactly one of these families of classes depending on the computational hardness (following directly from the implications above). For example, $\mathfrak{A}_\Pi$ includes planar graphs, $\mathfrak{E}_\Pi$ includes grids, and $\mathfrak{J}_\Pi$ includes trees. (The class NP-intermediate (NPI) contains those problems in NP which are neither in P nor NP-hard, unless P=NP.)

▶ **Observation 3.** *Let $\mathcal{G}$ be a hereditary graph class (i.e. $\mathcal{G} = \mathcal{G}_I$). Then $\mathcal{G}$ necessarily belongs to one of $\mathfrak{A} \cup \mathfrak{D} \cup \mathfrak{J}$ as a consequence of Equations* (1)–(4).

Our principal contributions are constructive proofs that each of these families of classes is nonempty. The intention is that these preliminary findings may help to motivate further research (see Section 6).

$\mathcal{G}$ is in ...	DomSet($\mathcal{G}$) is ...	ParDom($\mathcal{G}$) is ...	DomSet($\mathcal{G}_I$) is ...	Example member
$\mathfrak{A}$	NPc	NPc	NPc	planar graphs
$\mathfrak{B}$	NPI	NPc	NPc	$\mathcal{B}$ (Sec. 5.3)
$\mathfrak{C}$	NPI	NPI	NPc	$\mathcal{C}$ (Sec. 5.3)
$\mathfrak{D}$	NPI	NPI	NPI	$\mathcal{D}$ (Sec. 5.2)
$\mathfrak{E}$	P	NPc	NPc	grids [1]
$\mathfrak{F}$	P	NPI	NPc	$\mathcal{F}$ (Sec. 5.3)
$\mathfrak{G}$	P	NPI	NPI	$\mathcal{G}$ (Sec. 5.3)
$\mathfrak{H}$	P	P	NPc	$\mathcal{H}$ (Sec. 5.2)
$\mathfrak{I}$	P	P	NPI	$\mathcal{I}$ (Sec. 5.2)
$\mathfrak{J}$	P	P	P	trees

Table 1 The definition of various families of graph classes. We prove that all families are nonempty by constructing a member class for each.

5 Results

5.1 Tools

For ease, we shall denote $\mathcal{A}$ the class of general graphs, $\mathcal{E}$ the class of grids, and $\mathcal{J}$ the class of trees (as these belong to $\mathfrak{A}, \mathfrak{E}, \mathfrak{J}$ respectively).

▶ **Definition 4** (Leafing of a graph or graph class)**.** *Where G is a graph class, the* leafing *of G is the graph $L(G)$ obtained by creating, for each vertex v in G, a new vertex $l(v)$ adjacent only to v. (This graph may also be defined as the Corona product of G and K_1.) Where $\mathcal{G}$ is a graph class, the leafing of $\mathcal{G}$ is the class $L(\mathcal{G}) = \{L(G) : G \in \mathcal{G}\}$.*

▶ **Definition 5** (ℓ-subdivision)**.** *For $\ell \in \mathbb{N}$, the ℓ-subdivision of a graph G is a graph G' where each edge (u, v) is replaced by the path $\{u, uv_1, \ldots, uv_\ell, v\}$. The ℓ-subdivision of a graph class $\mathcal{G}$ is the $\{G' : G'$ is the ℓ-subdivision of some $G \in \mathcal{G}\}$*

Note that the set of neighbors of leaves in a leafed graph is a dominating set of minimum cardinality, yielding the following:

▶ **Lemma 6.** *For any graph class $\mathcal{G}$:*
- **(i)** *Dominating Set($L(\mathcal{G})$) is in P,*
- **(ii)** *Partial Domination($L(\mathcal{G})$) is at least as hard as Partial Domination($\mathcal{G}$), and*
- **(iii)** *Dominating Set($L(\mathcal{G})_I$) is at least as hard as Dominating Set($\mathcal{G}_I$)*

Proof. All three items are straightforward.

Item i follows from the fact that for any graph G of minimum degree one, $L(G)$ admits the set of vertices of degree one in $L(G)$ as a minimum dominating set.

Item ii can be obtained by observing that $L(G)$ is only polynomially larger than G (in fact, exactly double G in size) and that $L(G)[S] = G[S]$ for any $S \subseteq V(G)$.

Item iii is due to the fact that $\mathcal{G}_I \subseteq L(\mathcal{G})_I$ (note equality is possible - for example when $\mathcal{G}$ is the class planar graphs).

Open Q: is the converse of Item ii (resp. Item iii) true? Consider 2-subdivision graphs. ◀

▶ **Lemma 7.** *For any ℓ divisible by 3, the ℓ-subdivision of a graph G admits a dominating set of cardinality $|E(G)| \cdot \frac{l}{3} + k$ if and only if G has a vertex cover of size k.*

Proof. We first show the forward direction. Suppose G has a vertex cover C of cardinality k. We show that there is dominating set by constructing a set $C' \subseteq V(G') \setminus V(G)$ of cardinality $|E(G)|$; our dominating set will then be $C \cup C'$. For each edge $(u, v) \in E(G)$, C' contains $\{uv_i | i \mod 3 \equiv 0\}$ if $u \in C$ and $\{uv_i | i \mod 3 \equiv 1\}$ otherwise (in which case necessarily $v \in C$). It is straightforward to verify that $C \cup C'$ is then a dominating set of G' of the desired cardinality.

Now suppose that G' admits a dominating set D of cardinality $|E(G)| \cdot \frac{l}{3} + k$, then G admits a vertex cover of size k. Let D be a minimum dominating set of cardinality at most $|E(G)| \cdot \frac{l}{3} + k$. Note that necessarily $|D \cap V_{uv}| \geq \frac{\ell}{3}$ for each edge $(u, v) \in E(G)$. Moreover, we may compute a new minimum dominating set D^* as follows:

(a) If $v \in D$ and $u \notin D$, then let $v \in D^*$ and $\{uv_i | i \mod 3 \equiv 1\} \subset D^*$,

(b) Otherwise, let $u \in D^*$, and $\{uv_i | i \mod 3 \equiv 0\} \subset D^*$.

This construction cannot result in D^* being larger than D, and D^* is a dominating set of G' if D was. By construction $D^* \setminus V(G')$ is a vertex cover of G of the desired cardinality (since $|D^* \cap V(G')| = |E(G)| \cdot \frac{\ell}{3}$, and the result follows. ◀

We shall make use of the following conjecture, as is common in computational complexity:

▶ **Definition 8** (Exponential Time Hypothesis (ETH))**.** *3-SAT cannot be solved in $2^{o(n)}$ time.*

The ETH is strictly stronger than P=NP. Note that our line of investigation only makes sense assuming P≠ NP - though ETH is not strictly necessary, and we conjecture that our results could be obtained without relying on it.

▶ **Lemma 9.** *The class QP ∩ NP-complete is empty unless the ETH fails.*

Sketch of proof - likely textbook result**.** If there is such a Π, then reducing a 3-SAT instance ϕ to the problem is doable in polynomial time (by NP-completeness) and then solving the obtained instance of Π is doable in time $|I|^{\text{polylog}(|I|)}$ (by containment in QP).
Then solving ϕ "through" $|I|$ takes time at most $(|\phi|^a)^{\log^b(|\phi|^a)} \in |\phi|^{\text{polylog}|\phi|}$ (i.e., 3-SAT is in QP), so ETH fails. ◀

5.2 $\mathfrak{D}, \mathfrak{H}, \mathfrak{I}$ are nonempty.

Let $\mathcal{D}$ be the hereditary closure of the set of graphs obtained by subdividing n-vertex graphs $\lceil 2^{\sqrt{n}} \rceil_3$ times, where $\lceil x \rceil_3$ denotes the least multiple of three which is at least x.

$$\mathcal{D} := \left\{ G \text{ subdivided } \lceil 2^{\sqrt{|V(G)|}} \rceil_3 \text{ times } : G \in \mathcal{A} \right\}_I$$

▶ **Theorem 10.** *$\mathcal{D} \in \mathfrak{D}$.*

Proof. First observe that $\mathcal{D}$ is hereditary. Applying Observation 3, we need only show that Dominating Set($\mathcal{D}$) is NP-intermediate to show $\mathcal{D} \in \mathfrak{D}$.

▷ Claim 11. There can be no a polynomial-time algorithm for Dominating Set($\mathcal{D}$) unless the ETH fails.

Proof. Suppose there is an algorithm `A` solving Dominating Set($\mathcal{D}$) in polynomial time. That is, there is a constant c so that for any $G_D \in \mathcal{D}$ and $k_D \in \mathbb{N}$, $\mathtt{A}(G_D, k_d)$ runs in time $O(|V(D)|^c)$ and returns `true` if and only if $\gamma(G) \leq k_D$.

Now let G, k an instance of Vertex Cover with n vertices and m edges. Let G_D be the graph obtained by subdividing G $\lceil 2^{\sqrt{n}} \rceil_3$ times, and note that $G_D \in D$. Also, let $k_D = m \cdot \frac{\lceil 2^{\sqrt{n}} \rceil_3}{3} + k$, so that (applying Lemma 7) G_D admits a dominating set of size k_D if and only if G admits a vertex cover of size k. Applying the construction of G_D and then the algorithm `A` takes time $O(2^{c \log^2(n) \sqrt{n}})$ - yielding the desired contradition: a subexponential algorithm for Vertex Cover. ◁

▷ Claim 12. Dominating Set($\mathcal{D}$) is in QP.

Proof. Let $S(G) \subseteq V(G)$ be the set of supercubic vertices in a graph, (i.e., those with three or more neighbors) and denote $n_S = |S(G), n_V = |V(G)|$. Consider the following algorithm: given a graph G, iterate over all 2^{n_S} subsets of $S(G)$, and identify the minimum-size dominating set extends this subset using no other vertices from $S(G)$ in polynomial time. [Details omitted, hopefully sufficiently self-evident? A treewidth argument is overkill, but possible if necessary.] Clearly, the minimum dominating set for G will be among the dominating sets computed. Then the algorithm solves Dominating Set in $2^{n_S} + \mathrm{poly}(n_V)$ time. If G belongs to the class $\mathcal{D}$, then one of the following holds:

- $n_V \geq 2^{\sqrt{n_S}}$, and consequently $\log^2 n_V \geq n_S$, or
- every connected component of G is an isolated vertex, a subdivided path, or a subdivided claw graph $K_{1,3}$.

The first case corresponds to any graph in $\mathcal{D}$ wherein two vertices in $S(G)$ are connected by a path, since that path necessarily has length at least $2^{\sqrt{n_S}}$. For graphs in this case, the algorithm we gave runs in quasipolynomial time $2^{\log^2 n_V} + \mathrm{poly}(n_V)$. The second case corresponds to those graphs which are in $\mathcal{D}$ because of the hereditary closure operation - for these, Dominating Set is trivially solvable in polynomial time. Consequently, Dominating Set($\mathcal{D}$) is in the class QP. ◁

From Claim 11 we have that Dominating Set($\mathcal{D}$) is not solvable in polynomial time, and by applying Claim 12 and Lemma 9, it follows that Dominating Set($\mathcal{D}$) is not NP-complete. Consequently, Dominating Set($\mathcal{D}$) is NP-intermediate. ◀

We now apply this result to populate $\mathfrak{H}$ and $\mathfrak{I}$. First, let $\mathcal{H} := \left\{ G \oplus \overline{K_{2^{|V(G)|}}} : G \in \mathcal{A} \right\}$. That is, $\mathcal{H}$ is the family of graphs containing, for each graph G on n vertices (recall $\mathcal{A}$ is the class of general graphs), the disjoint union of G and 2^n isolated vertices.

▶ **Theorem 13.** $\mathcal{H} \in \mathfrak{H}$.

Proof. First observe that $\mathcal{H}_I = \mathcal{A}$, immediately yielding that Dominating Set($\mathcal{H}_I$) is NP-complete. Also note that, given any instance of Partial Domination($\mathcal{H}$), we may discard all singleton vertices from the instance (reducing the target k as appropriate if they are in the set T) to obtain a “kernel” instance of size logarithmic in the size of the original graph. We then may apply a brute-force algorithm to solving the kernel instance in time polynomial in the size of the original input (and exponential in the size of the kernel). Applying implication Equation (3) the result also holds for Dominating Set($\mathcal{H}$). ◀

We now combine both ideas above: let $\mathcal{I} := \left\{D \oplus \overline{K_{2^{|V(G)|}}} : D \in \mathcal{D}\right\}$. That is, $\mathcal{I}$ is the family of graphs containing, for each graph G on n vertices from $\mathcal{D}$, the disjoint union of G and 2^n isolated vertices.

▶ **Theorem 14.** $\mathcal{I} \in \mathfrak{I}$.

Proof - repetitive? First observe that $\mathcal{H}_I = \mathcal{D}$, so Dominating Set($\mathcal{H}_I$) is NP-intermediate (applying Theorem 10). Applying the same logic as in our proof of Theorem 13, any instance of Partial Domination($\mathcal{H}$) may be reduced to a "kernel" instance of size logarithmic in that of the original, and subsequently we may apply a brute-force algorithm to solving the kernel instance in time polynomial in the size of the original input. Again applying implication Equation (3) the result also holds for Dominating Set($\mathcal{I}$). ◀

5.3 $\mathfrak{B}, \mathfrak{C}, \mathfrak{F}, \mathfrak{G}$ are nonempty.

Using the results above, it becomes relatively straightforward to prove the following:
- $\mathfrak{B} \ni \mathcal{B} := \mathcal{E} \cup \mathcal{D}$,
- $\mathfrak{C} \ni \mathcal{D} \cup \mathcal{H}$,
- $\mathfrak{F} \ni \mathcal{F} := L(\mathcal{C})$ (applying Lemma 6), and
- $\mathfrak{G} \ni \mathcal{G} := L(\mathcal{D})$ (again applying Lemma 6)

6 Further questions

Without assuming P≠NP our problems become vacuous; nonetheless, it seems likely that many of our results could be proven without assuming ETH.

▶ **Question 1.** *Can we remove our reliance on ETH?*

Also, we note that the classes we construct in this paper are not especially natural, and expect that these have not been of previous interest.

▶ **Question 2.** *Which (if any)* natural *classes belong to* $\mathfrak{B}, \mathfrak{C}, \mathfrak{D}, \mathfrak{F}, \mathfrak{G}, \mathfrak{H}, \mathfrak{I}$*?*

One particularly interesting candidate is the family of hypercubes $\mathcal{Q}$.

▶ **Question 3.** *Is there a polynomial-time algorithm for Dominating Set($\mathcal{Q}$)?*

It is unlikely there is a practical polynomial-time algorithm for Dominating Set($\mathcal{H}$); even $\gamma(Q_{10})$ is unkown. See `https://oeis.org/A000983` and the discussion therein. On the other hand, it is provable (by leveraging Mahaney's Theorem [3]) that Dominating Set($\mathcal{H}$) is not NP-c unless P=NP. If the answer to Question 3 is negative, this would provide a first natural class belonging to $\mathfrak{B}$ (if the answer is positive, then $\mathcal{Q} \in \mathfrak{C}$, the same as grids).

References

1 David C. Kutner and Iain A. Stewart. Reconfigurable routing in data center networks, 2024. URL: https://arxiv.org/abs/2401.13359, arXiv:2401.13359.

2 David C. Kutner and Iain A. Stewart. Reconfigurable routing in data center networks. In Quentin Bramas, Arnaud Casteigts, and Kitty Meeks, editors, *Algorithmics of Wireless Networks*, pages 117–130, Cham, 2025. Springer Nature Switzerland.

3 Stephen R. Mahaney. Sparse complete sets for np: Solution of a conjecture of berman and hartmanis. *Journal of Computer and System Sciences*, 25(2):130–143, 1982. URL: https://www.sciencedirect.com/science/article/pii/0022000082900022, doi:10.1016/0022-0000(82)90002-2.