Discrete Applied Mathematics 99 (2000) 111–123

DISCRETE APPLIED MATHEMATICS

Domination and total domination on asteroidal triple-free graphs

Dieter Kratsch

Fakultät für Mathematik und Informatik, Friedrich-Schiller-Universität Jena, Universitaetshochhaus, 07740 Jena, Germany

Received 2 September 1997; received in revised form 10 February 1998; accepted 9 March 1999

Abstract

We present the first polynomial time algorithms for solving the NP-complete graph problems DOMINATING SET and TOTAL DOMINATING SET when restricted to asteroidal triple-free graphs. We also present algorithms to compute a minimum cardinality dominating set and a minimum cardinality total dominating set on a large superclass of the asteroidal triple-free graphs, namely the class of those graphs for which each connected component has a so-called dominating shortest path. Our algorithms can be implemented to run in time $O(n^6)$ on asteroidal-triple free graphs and in time $O(n^7)$ on graphs with dominating shortest path. © 2000 Elsevier Science B.V. All rights reserved.

1. Introduction

An asteroidal triple (AT in short) is a set of three vertices of a graph such that there is a path between any two of them avoiding the neighbourhood of the third one. Asteroidal triple-free graphs (AT-free graphs in short) form a large class of graphs containing interval, permutation, trapezoid and cocomparability graphs. Since 1989 AT-free graphs have been studied extensively by Corneil, Olariu and Stewart. They have published a collection of papers presenting many structural and algorithmic properties of AT-free graphs (see e.g. [9–11]).

Until 1995 the knowledge on the algorithmic complexity of NP-complete graph problems when restricted to AT-free graphs was relatively small compared to other graph classes. For example, some problems had been known to remain NP-complete when restricted to complements of bipartite graphs, a subclass of the AT-free graphs, among them TREEWIDTH, PATHWIDTH and MINIMUM FILL-IN [1,31]. Recently, it has been shown

E-mail address: kratsch@minet.uni-jena.de (D. Kratsch)

PII: S0166-218X(99)00128-6

that the problems INDEPENDENT SET and INDEPENDENT DOMINATING SET can be solved by $O(n^4)$ algorithms on AT-free graphs, while CLIQUE and PARTITION INTO CLIQUES remain NP-complete when restricted to AT-free graphs [6].

Domination problems have been considered extensively in the past 20 years both in Graph Theory and in Algorithmic Graph Theory. A Special Volume of *Discrete Mathematics* entitled "Topics on Domination" [23] appeared in 1990, a textbook and a monograph on domination in graphs [21,22] appeared recently. One of the reasons for the steady growth of the number of domination papers is the diversity of applications to both mathematical and real-world problems, in particular facility location problems. Furthermore, there is a great interest in finding polynomial time algorithms to solve various types of NP-complete domination problems when restricted to special classes of graphs.

Concerning the algorithmic complexity of domination-type problems on AT-free graphs the following is known. Balakrishnan, Rajaraman and Pandu Rangan presented $O(n^3)$ algorithms to compute a minimum cardinality connected dominating set and a minimum cardinality Steiner tree on AT-free graphs [2]. A linear time algorithm to compute a minimum cardinality connected dominating set on AT-free graphs with diameter greater than three has been given by Corneil et al. [10]. Furthermore, the existence problem DOMINATING CLIQUE, "Given a graph $G=(V,E)$, decide whether G has a dominating clique" is NP-complete on cocomparability graphs [27], and hence on AT-free graphs.

We consider the NP-complete graph problems DOMINATING SET and TOTAL DOMINATING SET, that remain NP-complete when restricted to any of the following graph classes: bipartite graphs, split graphs (see [13]) and circle graphs [25]. On the positive side, polynomial time algorithms have been designed for many graph classes (see [13,24]). For example, there are efficient algorithms to compute a minimum cardinality (total) dominating set for the following graph classes: interval graphs [8], strongly chordal graphs [7,17], cographs [12], permutation graphs [13,18,29,30], k-polygon graphs [16], cocomparability graphs [5,27], circular-arc graphs [8] and dually chordal graphs [4]. Particularly interesting with respect to this paper is a result of Breu and Kirkpatrick concerning a subclass of the class of AT-free graphs. They have given $O(nm^2)$ algorithms to compute a minimum cardinality dominating set and a minimum cardinality total dominating set on cocomparability graphs [5].

We present $O(n^6)$ algorithms to compute a minimum cardinality dominating set and a minimum cardinality total dominating set of a given AT-free graph. Thus we obtain the first polynomial time algorithms solving these problems on AT-free graphs. Our algorithms were also among the first polynomial time algorithms for any NP-complete problem, when restricted to AT-free graphs.

It is worth mentioning that our algorithms are designed for a large superclass of the class of AT-free graphs, that contains exactly those graphs having the structural property of AT-free graphs, which enables the design of our polynomial time algorithms for DOMINATING SET and TOTAL DOMINATING SET on AT-free graphs.

2. Preliminaries

We consider finite, undirected and simple graphs. Let $G=(V,E)$ be a graph. We denote by n the number of vertices of G and by $G[W]$ the subgraph *induced* by $W\subseteq V$. $N(v)=\{w\in V\colon \{v,w\}\in E\}$ is the *open neighbourhood* of $v\in V$, $N[v]=N(v)\cup\{v\}$ is the (*closed*) *neighbourhood* of $v\in V$ and $N[W]=\bigcup_{w\in W} N[w]$.

A sequence of vertices of a graph $G=(V,E)$ is a *path* $P=(u=x_0,x_1,\ldots,x_k=v)$ of G if $\{x_i,x_{i+1}\}\in E$ for all $i\in\{0,1,\ldots,k-1\}$. The *length* of a path $P=(u=x_0,x_1,\ldots,x_k=v)$ is defined to be k. The *distance* between two vertices u and v in G, denoted by $d_G(u,v)$, is the shortest length of a path between u and v. The *diameter* of a graph G is $\mathrm{diam}(G)=\max\{d_G(u,v)\colon u,v\in V\}$.

The set $D\subseteq V$ is a *dominating set* of $G=(V,E)$ if for every vertex $u\in V\setminus D$ there is a vertex $v\in D$ such that $\{u,v\}\in E$, i.e., $N[D]=V$. A dominating set D of G is a *total/connected/independent dominating set* of G, if $G[D]$ has no isolates, $G[D]$ is connected and D is an independent set, respectively. We denote the minimum cardinality of a dominating/total dominating/connected dominating set of a graph G by $\gamma(G)$, $\gamma_t(G)$ and $\gamma_{\mathrm{conn}}(G)$, respectively. The vertex $v\in V$ is a *dominating vertex* if $\{v\}$ is a dominating set and the edge $\{u,v\}\in E$ is a *dominating edge* if $\{u,v\}$ is a dominating set.

2.1. Special classes of graphs

Cocomparability graphs, permutation and interval graphs are well-known graph classes with many nice structural properties. For detailed information on these classes and on other special classes of graphs including structural properties and presentations of graphs in such classes we refer to [3,20].

Many NP-complete problems are solvable by polynomial time algorithms on such graph classes and often these algorithms are elegant, simple and practically efficient. Therefore, it is a natural intention to find larger graph classes still having at least some of the nice structural properties, say of the cocomparability graphs. This has been one of the motivations for the research of Corneil, Olariu and Stewart on AT-free graphs [9–11].

Definition. The vertices x, y and z of a graph $G=(V,E)$ form an *asteroidal triple* (AT in short) if $\{x,y,z\}$ is an independent set and for any two of these vertices there is a path between them that avoids the neighbourhood of the third. A graph G is said to be *asteroidal triple-free* (short AT-*free*) if it does not contain an asteroidal triple.

One of the major structural theorems on AT-free graphs is the dominating pair theorem.

Definition. (x,y) is a *dominating pair* of a graph $G=(V,E)$, if $x,y\in V$ and the vertex set of any path between x and y in G is a dominating set in G.

Theorem 1 (Corneil et al. [11]). *Any connected* AT*-free graph has a dominating pair.*

Our algorithms exploit a different structural property of AT-free graphs.

Definition. A path $P=(x=x_0,x_1,\dots,x_d=y)$ is a *dominating shortest path* (DSP in short) of a graph $G=(V,E)$, if $d_G(x,y)=d$ and $\{x_0,x_1,\dots,x_d\}$ is a dominating set of G.

Now, every connected AT-free graph G has a DSP, since any shortest path between the vertices of a dominating pair is a DSP. Furthermore, notice that any graph having a dominating vertex or a dominating edge has a DSP.

2.2. Approximation and domination

Every exact algorithm computing a minimum cardinality connected dominating set on a graph class $\mathfrak{G}$ may also be considered as an approximation algorithm. For any $G\in\mathfrak{G}$ the output D of the algorithm is a dominating set with $|D|\leqslant 3\gamma(G)$ and a total dominating set with $|D|\leqslant 2\gamma_t(G)$. This is an immediate consequence of theorems on the ratios of domination parameters [15,19] stating that for every graph G the following inequalities are fulfilled:

$$\gamma_{\mathrm{conn}}(G)\leqslant 3\gamma(G)-2 \quad [15],$$

$$\gamma_{\mathrm{conn}}(G)\leqslant 2\gamma_t(G)-2 \quad [19],$$

$$\gamma(G)\leqslant\gamma_t(G)\leqslant\gamma_{\mathrm{conn}}(G) \quad (\textit{Folclore}).$$

Thus, given a connected AT-free graph G, the linear time algorithm to compute a dominating pair (x,y) of G, presented in [10], can be applied to compute a connected dominating set D as the vertex set of an arbitrary shortest x,y-path. Then $|D|\leqslant\gamma_{\mathrm{conn}}(G)+2$ since any connected dominating set has cardinality at least $\mathrm{diam}(G)-1\geqslant|D|-2$ (see also [10]). This implies $|D|\leqslant 3\gamma(G)$ and $|D|\leqslant 2\gamma_t(G)$.

Hence there are fast algorithms to approximate the size of a minimum dominating set and the size of a minimum total dominating set on AT-free graphs.

Theorem 2. *There is a linear time algorithm with worst case performance ratio* 3 *to approximate the size of a minimum dominating set on* AT*-free graphs and there is a linear time algorithm with worst case performance ratio* 2 *to approximate the size of a minimum total dominating set on* AT*-free graphs.*

3. Small dominating sets in graphs with DSP

The theorems of the next two sections establish structural properties of graphs with DSP and of AT-free graphs that are crucial for the design and analysis of our algorithms.

They guarantee the existence of a "small" minimum cardinality dominating set D and a "small" minimum cardinality total dominating set T in the following sense: there is a BFS-tree of the graph such that the number of vertices of D (resp. T) in some consecutive levels of the BFS-tree is small.

Theorem 3. *Let $G=(V,E)$ be a graph with a* DSP *$P=(x=x_0,x_1,\dots,x_d=y)$. Let $H_0=\{x\}$, $H_1=N(x),\dots,H_i=\{w\in V\colon d_G(x,w)=i\},\dots,H_\ell=\{w\in V\colon d_G(x,w)=\ell\}$ be the* BFS*-levels of x. Then G has a minimum cardinality dominating set D and minimum cardinality total dominating set T such that*

$$\bigwedge_{i\in\{0,1,\dots,\ell\}}\ \bigwedge_{j\in\{0,1,\dots,\ell-i\}}\left|D\cap\bigcup_{s=i}^{i+j}H_s\right|\leqslant j+4, \tag{1}$$

$$\bigwedge_{i\in\{0,1,\dots,\ell\}}\ \bigwedge_{j\in\{0,1,\dots,\ell-i\}}\left|T\cap\bigcup_{s=i}^{i+j}H_s\right|\leqslant j+4. \tag{2}$$

Proof. First, we prove the existence of a minimum cardinality total dominating set with property (2) in each graph G having a DSP. Let T_r, r a positive integer, be any minimum cardinality total dominating set of G. Suppose T_r does not have property (2). We describe a procedure for constructing a new minimum cardinality total dominating set T_{r+1} of G.

$Q_r=\{(i,j)\colon |T_r\cap\bigcup_{s=i}^{i+j}H_s|\geqslant j+5\}\neq\emptyset$ since T_r does not have property (2). We choose $(i'_r,j'_r)\in Q_r$ such that first $i'_r=\min\{i\colon (i,j)\in Q_r\}$ and then $j'_r=\max\{j\colon (i'_r,j)\in Q_r\}$.

The properties of a BFS-tree ensure that any neighbour of a vertex in $T_r\cap(\bigcup_{s=i'_r}^{i'_r+j'_r}H_s)$ belongs to one of the levels $H_{i'_r-1},H_{i'_r},\dots,H_{i'_r+j'_r+1}$. Taking $A=\{x_{i'_r-2},x_{i'_r-1},\dots,x_{i'_r+j'_r+2}\}$, the fact that P is a DSP implies $N[A]\supseteq\bigcup_{s=i'_r-1}^{i'_r+j'_r+1}H_s$. Therefore, $T_{r+1}:=(T_r\setminus(\bigcup_{s=i'_r}^{i'_r+j'_r}H_s))\cup A$ is a dominating set of G. Furthermore, T_{r+1} is a total dominating set since $T_r\cap H_{i'_r-1}=\emptyset$ and $T_r\cap H_{j'_r+1}=\emptyset$ by the choice of (i'_r,j'_r). $|T_r\cap(\bigcup_{s=i'_r}^{i'_r+j'_r}H_s)|\geqslant j'_r+5$ and $|A|=j'_r+5$ implies $|T_r|\geqslant|T_{r+1}|$. Consequently, T_{r+1} is also a minimum cardinality total dominating set of G.

Notice that the boundary cases $i'_r\in\{0,1\}$ or $i'_r+j'_r\in\{\ell-1,\ell\}$ are impossible, since then the set A can be chosen as $A=\{x_0,x_1,\dots,x_{i'_r+j'_r+2}\}$, if $i'_r\in\{0,1\}$, and $A=\{x_{i'_r-2},x_{i'_r-1},\dots,x_d\}$, if $i'_r+j'_r\in\{\ell-1,\ell\}$. In both cases A contains less than j'_r+5 vertices. Thus the total dominating set T_{r+1} would be smaller than the minimum cardinality total dominating set T_r, a contradiction.

We call the replacement of T_r by T_{r+1} an *exchange step*. If T_{r+1} has property (2) then G has a minimum cardinality total dominating set with property (2). Otherwise, $Q_{r+1}=\{(i,j)\colon |T_{r+1}\cap\bigcup_{s=i}^{i+j}H_s|\geqslant j+5\}\neq\emptyset$. Suppose $(i,j)\in Q_{r+1}$ with $i\leqslant i_r$. Then $i+j\geqslant i'_r-2$, otherwise $(i,j)\in Q_r$, contradicting the choice of i'_r. By construction $|T_{r+1}\cap H_s|\geqslant 1$ for all $s\in\{i'_r-2,i'_r-1,\dots,i'_r+j'_r+1,i'_r+j'_r+2\}$. Thus $(i,j)\in Q_{r+1}$ with $i\leqslant i_r$ and $i+j\geqslant i'_r-2$ implies that there is a j' such that $(i,j')\in Q_{r+1}$ and $i+j'\geqslant i'_r+j'_r+2$. By the construction of T_{r+1}, this implies $|T_{r+1}\cap(\bigcup_{s=i}^{i+j'}H_s)|=|T_r\cap$

$(\bigcup_{s=i}^{i+j'} H_s)|$ and thus $(i,j') \in Q_r$, contradicting the choice of either i'_r or j'_r. Consequently $i'_{r+1} = \min\{i\colon (i,j) \in Q_{r+1}\} > i'_r$.

Therefore, each exchange step, that replaces a minimum cardinality total dominating set T_r by a new minimum cardinality total dominating set T_{r+1}, increases the minimum value of i for which there is a $j \in \{0,1,\dots,\ell - i\}$ with $|T_k \cap \bigcup_{s=i}^{i+j} H_s| \geqslant j+5$ for the current minimum cardinality total dominating set T_k. Hence starting with a minimum cardinality total dominating set T_1 of G we obtain a minimum cardinality total dominating set T having property (2) after at most d exchange steps.

Analogously, we can prove the existence of a minimum cardinality dominating set with property (1) in each graph G with DSP. Starting with an arbitrary minimum cardinality dominating set D_1 exactly the same exchange procedure can be used to obtain a minimum cardinality dominating set D having property (1) after at most d exchange steps. □

We show in Section 5 how to design an $O(n^7)$ algorithm to compute a minimum cardinality dominating set for graphs with DSP, using Theorem 3. It is worth mentioning that for our algorithmic purposes we actually need Theorem 3 only for the case of three consecutive BFS-levels.

4. Small dominating sets in AT-free graphs

The goal of this section is to obtain a theorem for connected AT-free graphs similar to Theorem 3 by improving the bounds for "small" minimum cardinality dominating sets and "small" minimum cardinality total dominating sets. This eventually allows us to improve the time bound of our algorithms when the inputs are restricted to AT-free graphs.

Theorem 4. *Let $G=(V,E)$ be a connected AT-free graph. There is a vertex $x \in V$ that can be determined in linear time with the following property. Let $H_0, H_1, \dots, H_\ell$ be the BFS-levels of x. Then there is a minimum cardinality dominating set D of G and a minimum cardinality total dominating set T of G such that*

$$\bigwedge_{i\in\{0,1,\dots,\ell\}} \; \bigwedge_{j\in\{0,1,\dots,\ell-i\}} \left| D \cap \bigcup_{s=i}^{i+j} H_s \right| \leqslant j+3, \tag{3}$$

$$\bigwedge_{i\in\{0,1,\dots,\ell\}} \; \bigwedge_{j\in\{0,1,\dots,\ell-i\}} \left| T \cap \bigcup_{s=i}^{i+j} H_s \right| \leqslant j+3. \tag{4}$$

Proof. There is a linear time algorithm to compute for any given connected AT-free graph G a path $P=(x=x_0,x_1,\dots,x_d=y)$ such that $x_i \in H_i$ for all $i \in \{0,1,\dots,d\}$, $V(P)=\{x_0,x_1,\dots,x_d\}$ is a dominating set of G and each vertex $z \in H_i$, $i \in \{0,1,\dots,\ell\}$ is adjacent to either x_{i-1} or x_i [26].

This algorithm first computes a dominating pair (x, y) of G, applying the linear time algorithm presented in [10]. Then the path P is constructed using the BFS-tree of x [26]. Since $V(P)$ is a dominating set each vertex $z\in H_i,\ i\in\{0,1,\dots,d\}$, is adjacent to $x_{i-1},\ x_i$ or x_{i+1}. Fortunately, the algorithm outputs a path P with the important property that each vertex $z\in H_i,\ i\in\{0,1,\dots,\ell\}$, is adjacent to either x_{i-1} or x_i.

This allows us to improve the bounds given in Theorem 3 for "small" dominating and total dominating sets in graphs with DSP. Although the proof does not differ much from the one of Theorem 3, for the sake of completeness we give the whole proof that there is a minimum cardinality dominating set D with property (3).

Let G be a connected AT-free graph. Let D_r, r a positive integer, be any minimum cardinality dominating set of G. Suppose D_r does not have property (3). Then $Q_r = \{(i,j){:}\ |D_r \cap \bigcup_{s=i}^{i+j} H_s|\geqslant j+4\} \neq \emptyset$. We choose $(i'_r, j'_r)\in Q_r$ such that first $i'_r = \min\{i{:}\ (i,j)\in Q_r\}$ and then $j'_r=\max\{j{:}\ (i'_r,j)\in Q_r\}$. (Analogously to the proof of Theorem 3, $i'_r\notin\{0,1\}$ and $i'_r+j'_r\notin\{\ell-1,\ell\}$.)

Every neighbour of a vertex in $D_r\cap(\bigcup_{s=i'_r}^{i'_r+j'_r} H_s)$ belongs to one of the levels $H_{i'_r-1},H_{i'_r},\dots,H_{i'_r+j'_r+1}$. We set $A=\{x_{i'_r-2},x_{i'_r-1},\dots,x_{i'_r+j'_r+1}\}$. Then $N[A]\supseteq\bigcup_{s=i'_r-1}^{i'_r+j'_r+1} H_s$, since every vertex $z\in H_k,\ k\in\{i'_r-1,\ i'_r,\dots,i'_r+j'_r+1\}$, is adjacent to x_{k-1} or x_k. Therefore, $D_{r+1} := (D_r\backslash(\bigcup_{s=i'_r}^{i'_r+j'_r} H_s))\cup A$ is a dominating set of G. $|D_r\cap(\bigcup_{s=i'_r}^{i'_r+j'_r} H_s)|\geqslant j'_r+4$ and $|A|=j'_r+4$ implies $|D_r|\geqslant|D_{r+1}|$. Consequently, D_{r+1} is also a minimum cardinality dominating set of G.

We call the replacement of D_r by D_{r+1} an *exchange step*. If D_{r+1} has property (3) then G has a minimum cardinality dominating set with property (3). Otherwise, $Q_{r+1}=\{(i,j){:}\ |D_{r+1}\cap\bigcup_{s=i}^{i+j} H_s|\geqslant j+4\}\neq\emptyset$. Suppose $(i,j)\in Q_{r+1}$ with $i\leqslant i_r$. Then $i+j\geqslant i'_r-2$, otherwise $(i,j)\in Q_r$, contradicting the choice of i'_r. By construction, $|D_{r+1}\cap H_s|\geqslant 1$ for all $s\in\{i'_r-2,\ i'_r-1,\dots,i'_r+j'_r,\ i'_r+j'_r+1\}$. Thus $(i,j)\in Q_{r+1}$ with $i\leqslant i_r$ and $i+j\geqslant i'_r-2$ implies that there is a j' such that $(i,j')\in Q_{r+1}$ and $i+j'\geqslant i'_r+j'_r+1$. By the construction of D_{r+1}, this implies $|D_{r+1}\cap(\bigcup_{s=i}^{i'+j'} H_s)| = |D_r\cap(\bigcup_{s=i}^{i+j'} H_s)|$ and thus $(i,j')\in Q_r$, contradicting the choice of either i'_r or j'_r. Consequently, $i'_{r+1}=\min\{i{:}\ (i,j)\in Q_{r+1}\} > i'_r$.

Therefore, starting with a minimum cardinality dominating set D_1 of G we obtain a minimum cardinality dominating set D having property (3) after $l\leqslant d$ exchange steps creating a sequence $D_1,D_2,\dots,D_{l-1},D_l$, since $1\leqslant i'_1 < i'_2 < \cdots < i'_{l-1}\leqslant d$.

The existence of a minimum cardinality total dominating set with property (4) can be shown by an analogous exchange argument. □

5. Domination

The key idea of our algorithms is to compute a certain type of dominating set by dynamic programming through the levels of a BFS-tree. It has been used, for example, in [14] to design an $O(n^3m)$ algorithm recognizing the so-called dominating diametral paths. Unfortunately, in general this natural approach does not lead to exact algorithms, not even to reasonable approximation algorithms.

Let us consider some details. For us a subsolution is a set $S \subseteq \bigcup_{j=0}^{i-1} H_j$, chosen during the dynamic programming up to a fixed level $i-1 \in \{1,2,\dots,\ell-1\}$. To collect the relevant information of any subsolution S it suffices to store the subset of those vertices of the subsolution, that belong to the last two levels, i.e. $S \cap (H_{i-2} \cup H_{i-1})$. Then it turns out that an upper bound on the maximum number of vertices that a minimum cardinality dominating set might have in any three consecutive BFS-levels is crucial for the running time of this type of algorithm. Notice that we have shown in the previous two sections that this number is at most 6 for graphs with DSP (Theorem 3) and at most 5 for connected AT-free graphs (Theorem 4).

First, we present a more general algorithm $mcds_w(G)$, w a fixed positive integer, that computes a dominating set of the given connected graph G. This algorithm could be applied to general graphs as a simple heuristic. However, the behaviour of this heuristic might be very bad. For example, if, for all dominating sets D of the input graph $G=(V,E)$ and for all vertices x of G, there are three consecutive BFS-levels of x such that S has more than w vertices in these three levels then the output of $mcds_w(G)$ is simply the trivial dominating set V.

If the input graph G has a vertex x and a minimum cardinality dominating set D such that at most w vertices of D belong to any three consecutive BFS-levels then $mcds_w(G)$ outputs a minimum cardinality dominating set of G.

algorithm $mcds_w(G)$
Input: *A connected graph* $G=(V,E)$.
Output: *A dominating set* $D \subseteq V$.
Initialize $D := V$;
FOR *all* $x \in V$ **DO**
  BEGIN
  Compute the BFS*-levels of vertex* x:
  $H_0 = \{x\},\ H_1 = N(x), \dots, H_\ell = \{u \in V\colon d_G(x,u) = l\}$;
  $i := 1$;
  Initialize the queue A_1 *to contain an ordered triple* $(S, S, val(S))$
  for all nonempty subsets S *of* $N[x]$ *satisfying* $val(S) := |S| \leqslant w$;
  WHILE $A_i \neq \emptyset$ **AND** $i < \ell$ **DO**
    BEGIN
    $i := i+1$;
    FOR *all triples* $(S, S', val(S'))$ *in the queue* A_{i-1} **DO**
      BEGIN
      FOR *every* $U \subseteq H_i$ *with* $|S \cup U| \leqslant w$ **DO**
      IF $N[S \cup U] \supseteq H_{i-1}$ **THEN**
        BEGIN
        $R := (S \cup U) \backslash H_{i-2}$;
        $R' := S' \cup U$;
        $val(R') = val(S') + |U|$;

IF *there is no triple in* A_i *with first entry* R
THEN *insert* $(R, R', val(R'))$ *in the queue* A_i;
IF *there is a triple* $(P, P', val(P'))$ *in* A_i *such that* $P = R$ **AND** $val(R') < val(P')$
THEN *replace* $(P, P', val(P'))$ *in* A_i *by* $(R, R', val(R'))$;
END;
END;
END;
Among all triples $(S, S', val(S'))$ *in the queue* A_ℓ, *that satisfy* $H_\ell \subseteq N[S]$, *determine one with minimum* $val(S')$, *say* $(B, B', val(B'))$;
IF $val(B') < |D|$ **THEN** $D := B'$;
END;
Output D.

Theorem 5. *Algorithm* $mcds_w(G)$ *computes in time* $O(n^{w+2})$ *a minimum cardinality dominating set of the given connected graph* $G = (V, E)$, *if* G *has a minimum cardinality dominating set* D *and a vertex* $x \in V$ *such that at most* w *vertices of* D *belong to any three consecutive* BFS-*levels of* x.

Proof. The running time of the part of the algorithm checking the BFS-levels of some fixed vertex x is $O(n^{w+1})$, since it is dominated by the time for the tests of all the subsets $S \cup U$ with $|S \cup U| \leqslant w$ that are contained in three consecutive BFS-levels of x. Notice that the amount of time per subset $S \cup U$ is $O(n)$ and that altogether there are $O(n^w)$ subsets $S \cup U$ to be tested when checking the BFS-levels of a fixed vertex x, since $|S \cup U| \leqslant w$. We emphasize that for avoiding duplicates the triples $(S, S', val(S'))$ are to be stored simultaneously in the corresponding queue A_i and also according to S in a w-dimensional array. (E.g. let $V = \{1, 2, \ldots, n\}$. Then $S = \{s_1, s_2, \ldots, s_k\}$ with $s_1 < s_2 < \cdots < s_k$ and $k \leqslant w$. Now, store $(S, S', val(S'))$ in the entry $(s_1, s_2, \ldots, s_k, s_{k+1}, \ldots, s_w)$ of the w-dimensional array taking $s_k = s_{k+1} = \cdots = s_w$.)

For any triple $(S, S', val(S'))$, the set S' represents a subsolution corresponding to S and $val(S')$. However, notice that only S and $val(S')$ are used in the dynamic programming. The main purpose of storing subsolutions S' is to facilitate finding a dominating set B' that corresponds to the value $val(B')$, which is the minimum cardinality of a dominating set, that has at most w vertices in any three consecutive BFS-levels of a fixed vertex x. (Of course this could also be done by a suitable pointer structure.)

We claim that for any $(S, S', val(S'))$ in the queue $A_i, \in \{1, 2, \ldots, \ell\}$: $S = S' \cap (H_{i-1} \cup H_i)$, $val(S') = |S'|$ and $N[S'] \supseteq \bigcup_{j=0}^{i-1} H_j$. This is true for $i = 1$. By the initialization of A_1, for all triples $(S, S', val(S))$ in A_1, $S = S'$ and $\emptyset \subset S \subseteq N[x]$. Thus $N[S] \supseteq H_0 = \{x\}$.

Suppose the claim is true for $i - 1 \in \{1, 2, \ldots, \ell - 1\}$. By the algorithm, the triple $(R, R', val(R'))$ is in A_i only if there is a triple $(S, S', val(S'))$ in A_{i-1} and a subset $U \subseteq H_i$ with $|S \cup U| \leqslant w$ and $N[S \cup U] \supseteq H_{i-1}$ such that $R = (S \cup U) \backslash H_{i-2}$, $R' = S' \cup U$ and $val(R') = val(S') + |U|$. Consequently, $R = R' \cap (H_{i-1} \cup H_i)$, $val(R') = |R'|$ and $N[R'] \supseteq \bigcup_{j=0}^{i-1} H_j$. This proves the claim.

Therefore, for any triple $(S,S',val(S))$ in A_ℓ with $N[S] \supseteq H_\ell$, S' is a dominating set of G. Consequently, for any minimum cardinality dominating set D of G such that at most w vertices of D belong to any three consecutive BFS-levels of x, there will be a triple $(D \cap (H_{\ell-1} \cup H_\ell), D', |D|)$ in A_ℓ such that $N[D] \supseteq H_\ell$, when the algorithm checks all BFS-levels of x. Hence the output of $mcds_w(G)$ is a minimum cardinality dominating set. □

As a matter of fact $mcds_w(G)$ is a heuristic of not much use in general. Fortunately, it turns into an exact polynomial time algorithm on some large graph classes for suitable choice of w. By Theorems 3 and 5, algorithm $mcds_6(G)$ computes a minimum cardinality dominating set of a graph with DSP in time $O(n^8)$. Observe that for given DSP $P=(x,x_1,x_2,\ldots,x_{d-1},y)$ of the input graph G it suffices to check only the BFS-levels of vertex x for finding a minimum cardinality dominating set and that this can be done in time $O(n^7)$. Given a graph for which each connected component has a dominating shortest path, we apply the above algorithm to each connected component, where a DSP of a connected component can be computed in time $O(n^3m)$ with a slight modification of an algorithm to decide whether a graph has a DSP of length diameter (called diametral path) given in [14]. Thus we obtain

Theorem 6. *There is an* $O(n^7)$ *time algorithm to compute a minimum cardinality dominating set of any given graph for which each connected component has a dominating shortest path.*

Analogously, Theorems 4 and 5 imply that algorithm $mcds_5(G)$ computes for a given connected AT-free graph a minimum cardinality dominating set. Furthermore, it suffices to compute a dominating pair (x,y) (see Section 4) and then to check the BFS-levels of vertex x only. Thus we obtain

Theorem 7. *There is an* $O(n^6)$ *time algorithm to compute a minimum cardinality dominating set for any given* AT*-free graph.*

6. Total domination

It is a matter of routine to design an $O(n^{w+2})$ time algorithm $mctds_w(G)$ similar to $mcds_w(G)$. Given a graph G without isolates, $mctds_w(G)$ computes a total dominating set of G.

Theorem 8. *Algorithm* $mctds_w(G)$ computes in time $O(n^{w+2})$ a minimum cardinality total dominating set of the given graph $G=(V,E)$, *if* G *has a minimum cardinality total dominating set* T and a vertex $x \in V$ *such that at most* w *vertices of* T *belong to any three consecutive* BFS*-levels of* x.

Combined with Theorems 3 and 4 we obtain

Theorem 9. *There is an* $O(n^7)$ *time algorithm to compute a minimum cardinality total dominating set for any given graph with* DSP *that has no isolates. There is an* $O(n^6)$ *time algorithm to compute a minimum cardinality total dominating set for any given* AT*-free graph that has no isolates.*

There is another way to obtain Theorem 9. For this purpose we consider false twins, i.e., two nonadjacent vertices of a graph with the same open neighbourhood. Then we say that a graph class $\mathcal{G}$ is closed under adding false twins if any graph G', obtained from a graph $G \in \mathcal{G}$ by choosing a vertex x of G and adding a new vertex x' such that the open neighbourhood of x' in G' is equal to $N(x)$, also belongs to $\mathcal{G}$. Now, it has been shown in [28] that the class of AT-free graphs is closed under adding false twins and it is not hard to show that the class of graphs with DSP is also closed under adding false twins. Then a linear time reduction between TOTAL DOMINATING SET and DOMINATING SET on graph classes closed under adding false twins (see [28]) ensures that Theorems 6 and 7 immediately imply Theorem 9.

7. Conclusions

We presented the first polynomial time algorithms solving the NP-complete problems DOMINATING SET and TOTAL DOMINATING SET on AT-free graphs. Our algorithms exploit the existence of a dominating shortest path in the input graph which guarantees the existence of a "small" minimum cardinality dominating/total dominating set. It is worth mentioning that, when considering dense graphs, the running time of the algorithms for AT-free graphs, i.e. $O(n^6)$, is not much worse than $O(nm^2)$, which is the running time of the best-known algorithms for cocomparability graphs [5].

By now the algorithmic complexity of the five most studied variants of the domination problem when restricted to AT-free graphs is known: DOMINATING SET, TOTAL DOMINATING SET, INDEPENDENT DOMINATING SET [6] and CONNECTED DOMINATION SET [2,10] are solvable in polynomial time, while DOMINATING CLIQUE [27] remains NP-complete. Nevertheless, there are some well-studied NP-complete graph problems for which the algorithmic complexity when restricted to AT-free graphs is unknown, in particular COLORING, HAMILTONIAN CIRCUIT and HAMILTONIAN PATH.

Contrary to well-known graph classes such as chordal, permutation and cocomparability graphs, we have not found yet any representation of AT-free graphs by a geometric intersection model, an elimination scheme of vertices or edges, a hypergraph model, etc. Such representations typically support the design of efficient algorithms. Hopefully, research on the above-mentioned problems may help us in finding such a representation for AT-free graphs.

References

[1] S. Arnborg, D.G. Corneil, A. Proskurowski, Complexity of finding embeddings in a k-tree, SIAM J. Algebraic Discrete Meth. 8 (1987) 277–284.

[2] H. Balakrishnan, R. Rajaraman, C. Pandu Rangan, Connected domination and Steiner set on asteroidal triple-free graphs, Proceedings of WADS'93, Lecture Notes in Computer Science, vol. 709, Springer, Berlin, 1993, pp. 131–141.

[3] A. Brandstädt, Special graph classes — a survey, Schriftenreihe des Fachbereichs Mathematik, SM-DU-199, Universität Duisburg Gesamthochschule, 1991.

[4] A. Brandstädt, V.D. Chepoi, F.F. Dragan, The algorithmic use of hypertree structure and maximum neighbourhood orderings, Proceedings of WG'94, Lecture Notes in Computer Science, vol. 903, Springer, Berlin, 1995, pp. 65–80.

[5] H. Breu, D.G. Kirkpatrick, Algorithms for dominating and Steiner set problems in cocomparability graphs, manuscript, 1993.

[6] H. Broersma, T. Kloks, D. Kratsch, H. Müller, Independent sets in asteroidal triple-free graphs, Proceedings of ICALP'97, Lecture Notes in Computer Science, vol. 1256, Springer, Berlin, 1997, pp. 760–770.

[7] G.J. Chang, Labelling algorithms for domination problems in sunfree chordal graphs, Discrete Appl. Math. 22 (1988) 21–34.

[8] M.S. Chang, Efficient algorithms for the domination problems on interval and circular-arc graphs, SIAM J. Comput. 27 (1998) 1671–1694.

[9] D.G. Corneil, S. Olariu, L. Stewart, A linear time algorithm to compute a dominating path in an AT-free graph, Inform. Process. Lett. 54 (1995) 253–258.

[10] D.G. Corneil, S. Olariu, L. Stewart, A linear time algorithm to compute dominating pairs in asteroidal triple-free graphs, Proceedings of ICALP'95, Lecture Notes in Computer Science, vol. 944, Springer, Berlin, 1995, pp. 292–302.

[11] D.G. Corneil, S. Olariu, L. Stewart, Asteroidal triple-free graphs, SIAM J. Discrete Math. 10 (1997) 399–430.

[12] D.G. Corneil, Y. Perl, Clustering and domination in perfect graphs, Discrete Appl. Math. 9 (1984) 27–39.

[13] D.G. Corneil, L. Stewart, Dominating sets in perfect graphs, Discrete Math. 86 (1990) 145–164.

[14] J.S. Deogun, D. Kratsch, Diametral path graphs, Proceedings of WG'95, Lecture Notes in Computer Science, vol. 1017, Springer, Berlin, 1995, pp. 344–357.

[15] P. Duchet, H. Meyniel, On Hadwiger's number and the stability number, Ann. Discrete Math. 13 (1982) 71–74.

[16] E.S. Elmallah, L.K. Stewart, Independence and domination in polygon graphs, Discrete Appl. Math. 44 (1993) 65–77.

[17] M. Farber, Domination, independent domination, and duality in strongly chordal graphs, Discrete Appl. Math. 7 (1984) 115–130.

[18] M. Farber, M. Keil, Domination in permutation graphs, J. Algorithms 6 (1985) 309–321.

[19] O. Favaron, D. Kratsch, Ratios of domination parameter, in: V. Kulli (Ed.), Advances in Graph Theory, Vishwa International, India, 1991, pp. 173–182.

[20] M.C. Golumbic, Algorithmic Graph Theory and Perfect Graphs, Academic Press, New York, 1980.

[21] T. Haynes, S.T. Hedetniemi, P. Slater, Fundamentals of Domination in Graphs, Marcel Dekker, New York, USA, 1998.

[22] T. Haynes, S.T. Hedetniemi, P. Slater (Eds.), Domination in Graphs: Advanced Topics, Marcel Dekker, New York, USA, 1998.

[23] S.T. Hedetniemi, R.C. Laskar (Eds.), Special Volume: Topics on Domination, Discrete Math. 86 (1990).

[24] D.S. Johnson, The NP-completeness column: an ongoing guide, J. Algorithms 5 (1984) 147–160.

[25] J.M. Keil, The complexity of domination problems in circle graphs, Discrete Appl. Math. 42 (1993) 51–63.

[26] T. Kloks, D. Kratsch, H. Müller, Approximating the bandwidth for AT-free graphs, Proceedings of ESA'95, Lecture Notes in Computer Science, vol. 979, Springer, Berlin, 1995, pp. 434–447.

[27] D. Kratsch, L. Stewart, Domination on cocomparability graphs, SIAM J. Discrete Math. 6 (1993) 400–417.

[28] D. Kratsch, L. Stewart, Total domination and transformation, Inform. Process. Lett. 63 (1997) 167–170.
[29] C. Rhee, Y.D. Liang, S.K. Dhall, S. Lakshmivarahan, An O($n + m$)-time algorithm for finding a minimum-weight dominating set in a permutation graph, SIAM J. Comput. 25 (1996) 404–419.
[30] K. Tsai, W.L. Hsu, Fast algorithms for the dominating set problem on permutation graphs, Algorithmica 9 (1993) 109–117.
[31] M. Yannakakis, Computing the minimum fill-in is NP-complete, SIAM J. Algebraic Discrete Meth. 2 (1981) 77–79.