Independent Sets in Asteroidal Triple-Free Graphs

Hajo Broersma[1] Ton Kloks[1] Dieter Kratsch[2] Haiko Müller[2]

[1] University of Twente
Faculty of Applied Mathematics
P.O. Box 217
7500 AE Enschede, the Netherlands
{H.J.Broersma,A.J.J.Kloks}@math.utwente.nl
[2] Fakultät für Mathematik und Informatik
Friedrich-Schiller-Universität
07740 Jena, Germany
{kratsch,hm}@minet.uni-jena.de

Abstract. An asteroidal triple is a set of three vertices such that there is a path between any pair of them avoiding the closed neighborhood of the third. A graph is called AT-free if it does not have an asteroidal triple. We show that there is an $O(n^2 \cdot (\overline{m} + 1))$ time algorithm to compute the maximum cardinality of an independent set for AT-free graphs, where n is the number of vertices and $\overline{m}$ is the number of non edges of the input graph. Furthermore we obtain $O(n^2 \cdot (\overline{m} + 1))$ time algorithms to solve the INDEPENDENT DOMINATING SET and the INDEPENDENT PERFECT DOMINATING SET problem on AT-free graphs. We also show how to adapt these algorithms such that they solve the corresponding problem for graphs with bounded asteroidal number in polynomial time. Finally we observe that the problems CLIQUE and PARTITION INTO CLIQUES remain NP-complete when restricted to AT-free graphs.

1 Introduction

Asteroidal triples were introduced in 1962 to characterize interval graphs as those chordal graphs that do not contain an asteroidal triple (short AT) [20]. Graphs not containing an AT are called asteroidal triple-free graphs (short AT-free graphs). They 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. [6, 7]). Further results on AT-free graphs were obtained in [18, 23].

Up to now 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. The problems TREEWIDTH, PATHWIDTH and MINIMUM FILL-IN remain NP-complete on AT-free graphs [1, 25]. On the other hand, domination-type problems like CONNECTED DOMINATING SET [7], DOMINATING SET [19] and TOTAL DOMINATING SET [19] can be solved by polynomial time algorithms for AT-free graphs. However there is a collection of classical NP-complete graph problems for which the algorithmic complexity when restricted to AT-free graphs was not known. Prominent representatives are INDEPENDENT SET, CLIQUE, GRAPH k-COLORABILITY, PARTITION INTO CLIQUES, HAMILTONIAN CIRCUIT and HAMILTONIAN PATH.

A crucial reason for the lack of progress in designing efficient algorithms for NP-complete problems on AT-free graphs seems to be that none of the typical representations, that are useful for the design of efficient algorithms on special graph classes, is known to exist for AT-free graphs. Contrary to well-known graph classes such as chordal, permutation and circular-arc graphs, AT-free graphs do not seem to have a representation by a geometric intersection model, an elimination scheme of vertices or edges, small separators, a small number of minimal separators etc. However it turns out that the design of all our algorithms is supported by a structural property of AT-free graphs, that can be obtained from the definition of AT-free graphs rather easily.

Our approach in this paper is similar to the one used to design algorithms for problems such as TREEWIDTH [14, 17] MINIMUM FILL-IN [17] and VERTEX RANKING [18] on AT-free graphs. However these algorithms have polynomial running time only under the additional constraint that the number of minimal separators is bounded by a polynomial in the number of vertices of the graph. (Notice that all three problems are NP-complete on AT-free graphs.) Technically, for the three different independent set problems in this paper, we are able to replace the set of all minimal separators, used in [14, 17, 18] – which might be 'too large' in size – by the 'small' set of all closed neighborhoods of the vertices of the graph.

Finding out the algorithmic complexity of INDEPENDENT SET on AT-free graphs is a challenging task. Besides the fact that INDEPENDENT SET is a classical and well-studied NP-complete problem, the problem is also interesting since, contrary to well-known subclasses of AT-free graphs such as cocomparability graphs, not all AT-free graphs are perfect. Thus the polynomial time algorithm for perfect graphs of Grötschel, Lovász and Schrijver [11] solving the INDEPENDENT SET problem does not apply to AT-free graphs.

We present the first polynomial time algorithm solving the NP-complete problem INDEPENDENT SET, when restricted to AT-free graphs. More precisely, our main result is the $O(n^2 \cdot (\overline{m}+1))$ algorithm to compute the maximum cardinality of an independent set in an AT-free graph. Furthermore we present an $O(n^2 \cdot (\overline{m}+1))$ time algorithm to solve the problem INDEPENDENT DOMINATING SET. A similar algorithm solves the problem INDEPENDENT PERFECT DOMINATING SET in time $O(n^2 \cdot (\overline{m}+1))$ [3]. We also observe that the problems CLIQUE and PARTITION INTO CLIQUES remain NP-complete when restricted to AT-free graphs.

A natural generalization of asteroidal triples are the so-called asteroidal sets. Structural results for asteroidal sets and algorithms for graphs with bounded asteroidal number were obtained in [15, 21]. Computing the asteroidal number (i.e., the maximum cardinality of an asteroidal set) turns out to be NP-complete in general, but solvable in polynomial time for many graph classes [16]. Furthermore the results for problems as TREEWIDTH and MINIMUM FILL-IN on AT-free graphs can be generalized to graphs with bounded asteroidal number [15]. We show how to adapt our algorithms to obtain polynomial time algorithms for graphs with bounded asteroidal number solving the problems INDEPENDENT SET, INDEPENDENT DOMINATING SET and INDEPENDENT PERFECT DOMINATING SET.

2 Preliminaries

For a graph $G = (V, E)$ we denote $|V|$ by n, $|E|$ by m and the number of edges of the complement of G, which is equal to the number of non edges of G, by $\overline{m}$.

Recall that an independent set in a graph G is a set of pairwise nonadjacent vertices. The independence number of a graph G denoted by $\alpha(G)$ is the maximum cardinality of an independent set in G.

For a graph $G = (V, E)$ and $W \subseteq V$, $G[W]$ denotes the subgraph of G induced by the vertices of W; we write $\alpha(W)$ for $\alpha(G[W])$. For convenience, for a vertex x of G we write $G - x$ instead of $G[V \setminus \{x\}]$. Analogously, for a subset $X \subseteq V$ we write $G - X$ instead of $G[V \setminus X]$. We consider components of a graph as (maximal connected) subgraphs as well as vertex subsets. For a vertex x of $G = (V, E)$, $N(x) = \{y \in V : \{x, y\} \in E\}$ is the neighborhood of x and $N[x] = N(x) \cup \{x\}$ is the closed neighborhood of x. For $W \subseteq V$, $N[W] = \bigcup_{x \in W} N[x]$.

A set $S \subseteq V$ is a separator of the graph $G = (V, E)$ if $G - S$ is disconnected.

Definition 1. Let $G = (V, E)$ be a graph. A set $\Omega \subseteq V$ is an *asteroidal set* if for every $x \in \Omega$ the set $\Omega \setminus \{x\}$ is contained in one component of $G - N[x]$. An asteroidal set with three vertices is called an *asteroidal triple* (short AT).

Notice that every asteroidal set is an independent set.

Remark. A triple $\{x, y, z\}$ of vertices of G is an asteroidal triple if and only if for every two of these vertices there is a path between them avoiding the closed neighborhood of the third.

Definition 2. A graph $G = (V, E)$ is called *asteroidal triple-free* (short AT-free) if G has no asteroidal triple.

It is well-known that the INDEPENDENT SET problem 'Given a graph G and a positive integer k, decide whether $\alpha(G) \geq k$', is NP-complete [9]. The problem remains NP-complete, even when restricted to cubic planar graphs [13]. Moreover the independence number is hard to approximate within a factor of $n^{1-\epsilon}$ for any constant $\epsilon > 0$ [12]. Despite this discouraging recent result on the complexity of approximation, the independence number can be computed in polynomial time on many special classes of graphs (see [13]). For example, the best known algorithm to compute the independence number of a cocomparability graph has running time $O(n + m)$ [24].

The main result of this paper is an $O(n^2 \cdot (\overline{m} + 1))$ algorithm to compute the maximum cardinality of an independent set in a given AT-free graph. The structural properties enabling the design of our algorithms are given in the next three sections. In this extended abstract, we restrict ourselves to the cardinality case of the problems. Nevertheless our algorithms can be extended in a straightforward manner such that they solve the corresponding problems on graphs with real vertex weights (see [3]).

3 Intervals

Let $G = (V, E)$ be an AT-free graph, and let x and y be two distinct nonadjacent vertices of G. Throughout the paper we use $C^x(y)$ to denote the component of $G - N[x]$ containing y, and $r(x)$ to denote the number of components of $G - N[x]$.

Definition 3. A vertex $z \in V \setminus \{x, y\}$ is *between* x and y if x and z are in one component of $G - N[y]$ and y and z are in one component of $G - N[x]$.

Equivalently, z is between x and y in G if there is an x, z-path avoiding $N[y]$ and there is a y, z-path avoiding $N[x]$.

Definition 4. The *interval* $I = I(x, y)$ of G is the set of all vertices of G that are between x and y.

Thus $I(x, y) = C^x(y) \cap C^y(x)$.

4 Splitting intervals

Let $G = (V, E)$ be an AT-free graph, let $I = I(x, y)$ be a nonempty interval of G and let $s \in I$. Let $I_1 = I(x, s)$ and $I_2 = I(s, y)$.

Lemma 5. *The vertices x and y are in different components of $G - N[s]$.*

Proof. Assume x and y would be in the same component of $G - N[s]$. Then there is an x, y-path avoiding $N[s]$. However $s \in I$ implies that there is an s, y-path avoiding $N[x]$ and an s, x-path avoiding $N[y]$. Thus $\{s, x, y\}$ is an AT of G, a contradiction. □

Corollary 6. $I_1 \cap I_2 = \emptyset$.

Proof. Assume $z \in I_1 \cap I_2$. Then $z \in I_1$ implies that there is a component C^s of $G - N[s]$ containing both x and z. Furthermore $z \in I_2$ implies that also $y \in C^s$, contradicting Lemma 5. □

Lemma 7. *$I_1 \subseteq I$ and $I_2 \subseteq I$.*

Proof. Let $z \in I_1$. Clearly $s \in I$ implies $s \in C^x(y)$. Thus $z \in I_1$ implies $z \in C^x(y)$. Clearly $z \in C^s(x)$ since $z \in I_1$. By Lemma 5, $C^s(x)$ is contained in a component of $G - N[y]$ and obviously this component contains x. This proves $z \in I$. Consequently $I_1 \subseteq I$.

$I_2 \subseteq I$ can be shown analogously. □

Theorem 8. *There exist components $C_1^s, C_2^s, \ldots, C_t^s$ of $G - N[s]$ such that*

$$I \setminus N[s] = I_1 \cup I_2 \cup \bigcup_{i=1}^{t} C_i^s.$$

Proof. By Lemma 7, we have $I_1 \subseteq I \setminus N[s]$ and $I_2 \subseteq I \setminus N[s]$. By Lemma 5, x and y belong to different components $C^s(x)$ and $C^s(y)$ of $G - N[s]$. Let $z \in I \setminus N[s]$.

Assume $z \in C^s(x)$. There is a z, y-path avoiding $N[x]$. This path must contain a vertex of $N[s]$, showing the existence of a z, s-path avoiding $N[x]$. Hence $z \in I_1$.

Similarly $z \in C^s(y)$ implies $z \in I_2$.

Assume $z \notin C^s(x)$ and $z \notin C^s(y)$. Since $z \notin N[s]$, z belongs to the component $C^s(z)$ of $G - N[s]$. For any vertex $p \in C^s(z)$, there is a p, z-path avoiding $N[x]$, since $C^s(z) \neq C^s(x)$. Since $z \in I$, there is a z, y-path avoiding $N[x]$. Hence there is also a p, y-path avoiding $N[x]$. This shows $C^s(z) \subseteq I \setminus N[s]$. □

Corollary 9. *Every component of $G[I \setminus (N[s] \cup I_1 \cup I_2)]$ is a component of $G - N[s]$.*

5 Splitting components

Let $G = (V, E)$ be an AT-free graph. Let C^x be a component of $G - N[x]$ and let y be a vertex of C^x. We study the components of the graph $C^x - N[y]$.

Theorem 10. *Let D be a component of $C^x - N[y]$. Then $N[D] \cap (N[x] \setminus N[y]) = \emptyset$ if and only if D is a component of $G - N[y]$.*

Proof. Let D be a component of $C^x - N[y]$ with $N[D] \cap (N[x] \setminus N[y]) = \emptyset$. Since no vertex of D has a neighbor in $N[x] \setminus N[y]$, D is a component of $G - N[y]$.

Now let $D \subseteq C^x$ be a component of $G - N[y]$. Then $N[D] \cap N[x] \subseteq N[y]$. □

Corollary 11. *Let B be a component of $C^x - N[y]$. Then $N[B] \cap (N[x] \setminus N[y]) \neq \emptyset$ if and only if $B \subseteq C^y(x)$.*

Theorem 12. *Let $B_1, \ldots, B_\ell$ denote the components of $C^x - N[y]$ that are contained in $C^y(x)$. Then $I(x,y) = \bigcup_{i=1}^{\ell} B_i$.*

Proof. Let $I = I(x,y)$. First we show that $B_i \subseteq I$ for every $i \in \{1, \ldots, \ell\}$. Let $z \in B_i$. There is an x,z-path avoiding $N[y]$, since some vertex in B_i has a neighbor in $N[x] \setminus N[y]$. Clearly, there is also a z,y-path avoiding $N[x]$, since z and y are both in C^x. This shows that $z \in I$. Consequently $\bigcup_{i=1}^{\ell} B_i \subseteq I$.

Suppose $z \in I \setminus \bigcup_{i=1}^{\ell} B_i$. Since $z \notin \bigcup_{i=1}^{\ell} B_i$, the component D of $C^x - N[y]$ containing z does not contain a vertex with a neighbor in $N[x] \setminus N[y]$. Thus $z \notin C^y(x)$, implying $z \notin I$, a contradiction. □

6 Computing the independence number

In this section we describe our algorithm to compute the independence number of an AT-free graph. The algorithm we propose uses dynamic programming on intervals and components. All intervals and all components are sorted according to nondecreasing number of vertices. Following this order, the algorithm determines the independence number of each component and of each interval using the formulas given in Lemmas 13, 14 and 15.

We start with an obvious lemma.

Lemma 13. *Let $G = (V, E)$ be any graph. Then*

$$\alpha(G) = 1 + \max_{x \in V} \Big(\sum_{i=1}^{r(x)} \alpha(C_i^x) \Big),$$

where $C_1^x, C_2^x, \ldots, C_{r(x)}^x$ are the components of $G - N[x]$.

Applying Lemma 13 to the decomposition given by Theorems 10 and 12, we obtain the following lemma.

Lemma 14. *Let $G = (V, E)$ be an AT-free graph. Let $x \in V$ and let C^x be a component of $G - N[x]$. Then*

$$\alpha(C^x) = 1 + \max_{y \in C^x} \Big(\alpha(I(x,y)) + \sum_i \alpha(D_i^y) \Big),$$

where the D_i^y's are the components of $G - N[y]$ contained in C^x.

Applying Lemma 13 to the decomposition given by Theorem 8, we obtain the following lemma.

Lemma 15. *Let $G = (V, E)$ be an AT-free graph. Let $I = I(x, y)$ be an interval of G. If $I = \emptyset$ then $\alpha(I) = 0$. Otherwise*

$$\alpha(I) = 1 + \max_{s \in I} \left(\alpha(I(x,s)) + \alpha(I(s,y)) + \sum_i \alpha(C_i^s)\right),$$

where the C_i^s's are the components of $G - N[s]$ contained in $I(x, y)$.

Remark. Notice that the components D_i^y and C_i^s as well as the intervals $I(x, s)$ and $I(s, y)$ on the right-hand side of the formulas in Lemma 14 and Lemma 15 are proper subsets of C^x and I, respectively. Hence $\alpha(C^x)$ (resp. $\alpha(I)$) can be computed by table look-up to components and intervals with a smaller number of vertices.

Consequently we obtain the following algorithm to compute the independence number $\alpha(G)$ for a given AT-free graph $G = (V, E)$, which is based on dynamic programming.

Step 1 For every $x \in V$ compute all components $C_1^x, C_2^x, \ldots, C_{r(x)}^x$ of $G - N[x]$.
Step 2 For every pair of nonadjacent vertices x and y compute the interval $I(x, y)$.
Step 3 Sort all the components and intervals according to nondecreasing number of vertices.
Step 4 Compute $\alpha(C)$ and $\alpha(I)$ for each component C and each interval I in the order of Step 3.
Step 5 Compute $\alpha(G)$.

Theorem 16. *There is an $O(n^2 \cdot (\overline{m} + 1))$ time algorithm to compute the independence number of a given AT-free graph.*

Proof. The correctness of our algorithm follows from the formulas of Lemmas 13, 14 and 15 as well as the order of the dynamic programming.

We show how to obtain the stated time complexity. Clearly, Step 1 can be implemented such that it takes $O(n(n+m))$ time using a linear time algorithm to compute the components of the graph $G - N[x]$ for each vertex x of G. For each component of $G - N[x]$, a sorted linked list of all its vertices and its number of vertices is stored. For all nonadjacent vertices x and y there is a pointer $P(x, y)$ to the list of $C^x(y)$. Thus in Step 2, an interval $I(x, y)$ can be computed using the fact that $I(x, y) = C^x(y) \cap C^y(x)$. Hence a sorted vertex list of $I(x, y)$ can be computed in time $O(n)$ for each interval. Consequently the overall time bound for Step 2 is $O(n \cdot (\overline{m} + 1))$. There are at most n^2 components and at most n^2 intervals and each has at most n vertices. Thus using the linear time sorting algorithm bucket sort, Step 3 can be done in time $O(n^2)$.

The bottleneck for the time complexity of our algorithm is Step 4. First consider a component C^x of $G - N[x]$ and a vertex $y \in C^x$. We need to compute the components of $G - N[y]$ that are contained in C^x. Each component D of $G - N[y]$ except $C^y(x)$ is contained in C^x if and only if $D \cap C^x \neq \emptyset$. Thus the components D of $G - N[y]$ with $D \subseteq C^x$ are exactly those components of $G - N[y]$ addressed by $P(y, z)$ for some $z \in C^x$. Thus all such components can be found in time $O(|C^x|)$ for fixed vertices x and $y \in C^x$. Hence the computation of $\alpha(C)$ for all components C takes time $\sum_{\{x,y\} \notin E} O(|C^x(y)|) = O(n \cdot (\overline{m} + 1))$.

Now consider an interval $I = I(x, y)$, and a vertex $s \in I$. We need to add up the independence numbers of the components C_i^s of $G - N[s]$ that are contained in I. The components of $G - N[y]$ that are contained in I are exactly those components addressed by $P(y, z)$ for some $z \in I$, except $C^s(x)$ and $C^s(y)$. Thus all such components can be found in time $O(|I(x, y)|)$ for a fixed interval $I(x, y)$ and $s \in I(x, y)$. Hence the computation of $\alpha(I)$ for all intervals I takes time $\sum_{\{x,y\} \notin E} \sum_{s \in I(x,y)} O(|I(x, y)|) = O(n^2 \cdot (\overline{m} + 1))$.

Clearly Step 5 can be done in $O(n^2)$ time. Thus the running time of our algorithm is $O(n^2 \cdot (\overline{m} + 1))$. □

7 Independent domination

The approach used to design the presented polynomial time algorithm to compute the independence number for AT-free graphs can also be used to obtain a polynomial time algorithm solving the INDEPENDENT DOMINATING SET problem on AT-free graphs. The best known algorithm to solve the weighted version of the problem on cocomparability graphs has running time $O(n^{2.376})$ [4].

Definition 17. Let $G = (V, E)$ be a graph. Then $S \subseteq V$ is a *dominating set* of G if every vertex of $V \setminus S$ has a neighbor in S. A dominating set $S \subseteq V$ is an *independent dominating set* of G if S is an independent set.

We denote by $\gamma_i(G)$ the minimum cardinality of an independent dominating set of the graph G. Given an AT-free graph G, our next algorithm computes $\gamma_i(G)$. It works very similar to the algorithm of the previous section.

We present only the formulas used in Step 4 and 5 of the algorithm (which are similar to those in Lemma 13, Lemma 14 and Lemma 15).

Lemma 18. *Let $G = (V, E)$ be a graph. Then*

$$\gamma_i(G) = 1 + \min_{x \in V} \Big(\sum_{j=1}^{r(x)} \gamma_i(C_j^x) \Big),$$

where $C_1^x, C_2^x, \ldots, C_{r(x)}^x$ are the components of $G - N[x]$.

Lemma 19. *Let $G = (V, E)$ be an AT-free graph. Let $x \in V$ and let C^x be a component of $G - N[x]$. Then*

$$\gamma_i(C^x) = 1 + \min_{y \in C^x} \Big(\gamma_i(I(x, y)) + \sum_j \gamma_i(D_j^y) \Big),$$

where the D_j^y's are the components of $G - N[y]$ contained in C^x.

Lemma 20. *Let $G = (V, E)$ be an AT-free graph. Let $I = I(x, y)$ be an interval. If $I = \emptyset$ then $\gamma_i(I) = 0$. Otherwise*

$$\gamma_i(I) = 1 + \min_{s \in I} \Big(\gamma_i(I(x, s)) + \gamma_i(I(s, y)) + \sum_j \gamma_i(C_j^s) \Big),$$

where the C_j^s's are the components of $G - N[s]$ contained in $I(x, y)$.

Design and analysis of the algorithm is done similar to the previous section. We obtain the following theorem.

Theorem 21. *There exists an $O(n^2 \cdot (\overline{m} + 1))$ time algorithm to compute the independence domination number γ_i of a given AT-free graph.*

In the full version [3] we also show how to obtain an $O(n^2 \cdot (\overline{m} + 1))$ algorithm to compute a minimum cardinality independent perfect dominating set for AT-free graphs.

8 Bounded asteroidal number

In this section we show that the independence number of graphs with bounded asteroidal number can be computed in polynomial time.

Definition 22. The *asteroidal number* of a graph G is the maximum cardinality of an asteroidal set in G.

Hence a graph is AT-free if and only if its asteroidal number is at most two. Furthermore the asteroidal number of a graph G is bounded by $\alpha(G)$, since every asteroidal set is an independent set.

Definition 23. Let Ω be an asteroidal set of G. The *lump* $L(\Omega)$ is the set of vertices v such that for all $x \in \Omega$ there is a component of $G - N[x]$ containing v and $\Omega \setminus \{x\}$.

Let $\Omega = \{x_1, \ldots, x_\kappa\}$ be an asteroidal set of cardinality $\kappa \geq 2$ and consider the lump $L = L(\Omega)$.

Let s be an arbitrary vertex in L. In this section we show how $N[s]$ splits the lump analogous to Theorem 8.

Consider the components of $G - N[s]$. These components partition Ω into sets $\Omega_1, \ldots, \Omega_\tau$, where each Ω_i is a maximal subset of Ω contained in a component of $G - N[s]$.

Lemma 24. *For each $i = 1, \ldots, \tau$, the set $\Omega_i^* = \Omega_i \cup \{s\}$ is an asteroidal set in G.*

Proof. Consider $x \in \Omega_i$. Then, by definition, $\Omega \setminus \{x\}$ and s are contained in one component of $G - N[x]$. Hence, $\Omega_i^* \setminus \{x\}$ is contained in one component of $G - N[x]$. This proves the claim. □

Lemma 25. *Let $z \in L$ be in some component C^* of $G - N[s]$ that contains no vertices of Ω. Then $C^* \subseteq L$.*

Proof. Let $p \in C^* \setminus \{z\}$. There is a p, z-path avoiding $N[x]$ for any vertex $x \in \Omega$. This proves the claim. □

First we consider the case where $\tau = 1$, i.e., where Ω is in one component of $G - N[s]$. Then $\Omega \cup \{s\}$ is an asteroidal set.

Lemma 26. *If Ω is contained in one component C of $G - N[s]$, then $L(\Omega \cup \{s\}) = L \cap C$.*

Proof. Clearly $L(\Omega \cup \{s\}) \subseteq L \cap C$. Let $z \in L \cap C$ and consider a vertex $x \in \Omega$. Clearly, there is an x, z-path avoiding $N[s]$, since z and x are in the component C of $G - N[s]$. Hence z is in the component of Ω of $G - N[s]$. Consider any other vertex $y \in \Omega$. (Such vertices exist since $|\Omega| \geq 2$). There exists a z, y-path avoiding $N[x]$ since $z \in L$. But also, there exists a y, s-path avoiding $N[x]$ since $\Omega \cup \{s\}$ is an asteroidal set. Hence z is in the component of $(\Omega \cup \{s\}) \setminus \{x\}$ of $G - N[x]$. □

Now we consider the case where $\tau > 1$. Let $L_i = L(\Omega_i \cup \{s\})$ for $i = 1, \ldots, \tau$. Clearly, $L_i \cap L_j = \emptyset$ for every $i \neq j$.

Lemma 27. *Assume $\tau > 1$ and let C be the component of $G - N[s]$ containing Ω_i. Then $L_i = L \cap C$.*

Proof. First let $z \in L \cap C$. Then for all x and y in Ω_i there is a z, x-path avoiding $N[s]$ since $z \in C$ (showing that z and Ω_i are in one component of $G - N[s]$), and there is a z, x-path avoiding $N[y]$ since $z \in L$. For $y' \in \Omega_j$ for any $j \neq i$ there is a z, y'-path avoiding $N[x]$, since $z \in L$. Such a path contains a vertex of $N[s]$, and consequently there is a z, s-path avoiding $N[x]$. This shows that z, s and $\Omega_i \setminus \{x\}$ are in one component of $G - N[x]$ and hence $L \cap C \subseteq L_i$.

Now let $z \in L_i$. This clearly implies $z \in C$. For a vertex $y \in \Omega_j$, $j \neq i$, s and the set $\Omega \setminus \{y\}$ are in one component of $G - N[y]$ since $s \in L$. There is an s, z-path avoiding $N[y]$ since y and z belong to different components of $G - N[s]$. Consequently, z and $\Omega \setminus \{y\}$ are in one component of $G - N[y]$.

For a vertex $x \in \Omega_i$, there is a component of $G - N[x]$ containing s and $\Omega \setminus \{x\}$, since $s \in L$. Since $z \in L_i$, there is an s, z-path avoiding $N[x]$. Hence also z is in this component of $G - N[x]$ and therefore $L_i \subseteq L \cap C$. □

Theorem 28. *There exist components $C_1, \ldots, C_t$ of $G - N[s]$ which contain no vertex of Ω such that*

$$L \setminus N[s] = \bigcup_{i=1}^{t} C_i \cup \bigcup_{j=1}^{\tau} L_j.$$

Proof. Let $C_1, \ldots, C_t$ be the components of $G - N[s]$ which contain a vertex of L but no vertex of Ω. Then by Lemma 25 we have $\bigcup_{i=1}^{t} C_i \subseteq L \setminus N[s]$, and by Lemmas 26 and 27 we have $\bigcup_{j=1}^{\tau} L_j \subseteq L \setminus N[s]$.

Now let $l \in L \setminus N[s]$. If l is in a component containing Ω_i, $1 \leq i \leq \tau$, then $l \in L_i$ by Lemma 26 or 27. Otherwise there is an index i, $1 \leq i \leq t$ such that $l \in C_i$. This completes the proof. □

Theorem 28 enables us to generalize Lemmas 15 and 20 in the following way.

Lemma 29. *Let $L = L(\Omega)$ be a lump of G. If $L = \emptyset$ then $\alpha(L) = \gamma_i(L) = 0$. Otherwise*

$$\alpha(L) = 1 + \max_{s \in L} \Big(\sum_{j=1}^{t} \alpha(C_j) + \sum_{i=1}^{\tau} \alpha(L_i) \Big),$$

$$\gamma_i(L) = 1 + \min_{s \in L} \Big(\sum_{j=1}^{t} \gamma_i(C_j) + \sum_{k=1}^{\tau} \gamma_i(L_k) \Big),$$

where $C_1, \ldots, C_t$ are the components of $G - N[s]$ which contain no vertex of Ω, $L_1, \ldots, L_\tau$ are the lumps $L(\Omega_i + s)$ as used in Lemma 24.

Together with Lemmas 13 and 14, 18 and 19, the formulas of Lemma 29 lead to recursive algorithms computing $\alpha(G)$ and $\gamma_i(G)$ for a graph G. For any positive integer k, these algorithms can be implemented to run in time $O(n^{k+2})$ for all graphs with asteroidal number at most k. Analogously to the proof of Theorem 16, the time complexity is now dominated by the term $\sum_\Omega \sum_{s \in L(\Omega)} O(|L(\Omega)|) = O(n^{k+2})$, where the sum is taken over all asteroidal sets Ω of G and all $s \in L(\Omega)$.

As before, our algorithms for graphs with a bounded asteroidal number can be extended to the weighted cases of the problems and the corresponding algorithms have the same timebounds.

9 Conclusions

In this paper we have shown that the independence number as well as the independence domination number of an AT-free graph can be computed in time $O(n^2 \cdot (\overline{m} + 1))$. The same approach can be used to obtain an $O(n^2 \cdot (\overline{m} + 1))$ algorithm to solve the INDEPENDENT PERFECT DOMINATING SET problem on AT-free graphs. We have shown how to adapt the algorithm computing the independence number in such a way that the new algorithm computes the independence number of a graph with a bounded asteroidal number in polynomial time.

In the full version [3] we show how to extend our algorithms for the problems INDEPENDENT SET and INDEPENDENT DOMINATING SET to AT-free graphs with real vertex weights. Both algorithms run in time $O(n^2 \cdot (\overline{m} + 1))$. Furthermore our algorithms can also be modified such that they compute a maximum weight independent set and a minimum weight independent dominating set in time $O(n^2 \cdot (\overline{m} + 1))$.

Contrary to the independent set problems considered so far, the NP-complete graph problems CLIQUE and PARTITION INTO CLIQUES, that are closely related to INDEPENDENT SET, both remain NP-complete when restricted to the class of AT-free graphs. Concerning CLIQUE recall that Poljak has shown that INDEPENDENT SET remains NP-complete on triangle-free graphs [9]. Consequently CLIQUE remains NP-complete on graphs with independence number at most two, and thus on AT-free graphs. Similarly, it follows from a recent result due to Maffray and Preissman (showing that GRAPH k-COLORABILITY remains NP-complete when restricted to triangle-free graphs [22]), that the problem PARTITION INTO CLIQUES remains NP-complete on AT-free graphs.

Consequently CLIQUE and PARTITION INTO CLIQUES are the first NP-complete graph problems (known to us) which are NP-complete on AT-free graphs, but solvable in polynomial time on the class of cocomparability graphs. The latter graph class is the largest well-studied subclass of AT-free graphs which is also a class of perfect graphs.

It would be interesting to find out the algorithmic complexity of the following well-known NP-complete graph problems when restricted to AT-free graphs: GRAPH k-COLORABILITY, HAMILTONIAN CIRCUIT, HAMILTONIAN PATH. These three problems are all known to have polynomial time algorithms for cocomparability graphs [8, 10].

References

1. Arnborg, S., D. G. Corneil and A. Proskurowski, Complexity of finding embeddings in a k-tree, *SIAM J. Alg. Disc. Meth.* **8** (1987), pp. 277–284.
2. Brandstädt, A., Special graph classes – A survey, Schriftenreihe des Fachbereichs Mathematik, SM-DU-199, Universität Duisburg Gesamthochschule, 1991.
3. Broersma, H. J., T. Kloks, D. Kratsch and H. Müller, Independent sets in asteroidal triple-free graphs, Memorandum No. 1359, Faculty of Applied Mathematics, University of Twente, Enschede, The Netherlands, 1996.
4. Breu, H. and D. G. Kirkpatrick, Algorithms for domination and Steiner tree problems in cocomparability graphs, Manuscript 1993.
5. Chang, M. S., Weighted domination on cocomparability graphs, *Proceedings of ISAAC'95*, Springer-Verlag, LNCS 1004, 1996, pp. 122–131.
6. Corneil, D. G., S. Olariu and L. Stewart, The linear structure of graphs: Asteroidal triple-free graphs, *Proceedings of WG'93*, Springer-Verlag, LNCS 790, 1994, pp. 211–224.
7. Corneil, D. G., S. Olariu and L. Stewart, A linear time algorithm to compute dominating pairs in asteroidal triple-free graphs, *Proceedings of ICALP'95*, Springer-Verlag, LNCS 944, 1995, pp. 292–302.
8. Deogun, J. S. and G. Steiner, Polynomial algorithms for hamiltonian cycle in cocomparability graphs, *SIAM J. Comput.* **23** (1994), pp. 520–552.
9. Garey, M. R. and D. S. Johnson, *Computers and Intractability: A guide to the theory of NP-completeness*, Freeman, San Francisco, 1979.
10. Golumbic, M. C., *Algorithmic graph theory and perfect graphs*, Academic Press, New York, 1980.
11. Grötschel, M., L. Lovász and A. Schrijver, Polynomial algorithms for perfect graphs, *Annals of Discrete Mathematics* **21** (1984), pp. 325–356.
12. Hastad, J., Clique is hard to approximate within $n^{1-\epsilon}$, to appear in the *Proceedings of FOCS'96*.
13. Johnson, D. S., The NP-completeness column: An ongoing guide, *J. Algorithms* **6** (1985), pp. 434–451.
14. Kloks, T., *Treewidth – Computations and Approximations*, Springer-Verlag, LNCS 842, 1994.
15. Kloks, T., D. Kratsch and H. Müller, A generalization of AT-free graphs and some algorithmic results, Manuscript 1996.
16. Kloks, T., D. Kratsch and H. Müller, Asteroidal sets in graphs, Memorandum No. 1347, Faculty of Applied Mathematics, University of Twente, Enschede, The Netherlands, 1996.
17. Kloks, T., D. Kratsch and J. Spinrad, On treewidth and minimum fill-in of asteroidal triple-free graphs, to appear in *Theoretical Computer Science* **175** (1997).
18. Kloks, T., H. Müller and C. K. Wong, Vertex ranking of asteroidal triple-free graphs, *Proceedings of ISAAC'96*, Springer-Verlag, LNCS 1178, 1996, pp. 174-182.
19. Kratsch, D., Domination and total domination on asteroidal triple-free graphs, Forschungsergebnisse Math/Inf/96/25, FSU Jena, Germany, 1996.
20. Lekkerkerker, C. G. and J. Ch. Boland, Representation of a finite graph by a set of intervals on the real line, *Fund. Math.* **51** (1962), pp. 45–64.
21. Lin, I. J., T. A. McKee and D. B. West, Leafage of chordal graphs, Manuscript 1994.
22. Maffray, F. and M. Preissman, On the NP-completeness of the k-colorability problem for triangle-free graphs, *Discrete Mathematics* **162** (1996), pp. 313–317.
23. Möhring, R. H., Triangulating graphs without asteroidal triples, *Discrete Applied Mathematics*, **64** (1996), pp. 281–287.
24. McConnell, R. M. and J. P. Spinrad, Modular decomposition and transitive orientation, Manuscript 1995.
25. Yannakakis, M., Computing the minimum fill-in is NP-complete, *SIAM J. Alg. Disc. Meth.* **2** (1981), pp. 77–79.