Algorithms for Finding Clique-Transversals of Graphs

Guillermo Durán[1]
Universidad de Chile, Departamento de Ingeniería Industrial
Santiago, Chile
gduran@dii.uchile.cl

Min Chih Lin[2]
Universidad de Buenos Aires, Departamento de Computación
Buenos Aires, Argentina
oscarlin@dc.uba.ar

Sergio Mera[3]
Universidad de Buenos Aires, Departamento de Computación
Buenos Aires, Argentina
smera@dc.uba.ar

Jayme L. Szwarcfiter[4]
Universidade Federal do Rio de Janeiro, I. Matemática, NCE and COPPE
Rio de Janeiro, RJ, Brasil
jayme@nce.ufrj.br

Contact Author: Jayme L Szwarcfiter
NCE-Universidade Federal do Rio de Janeiro
Caixa Postal 2324
20001-970 Rio de Janeiro, RJ
tel: (55)(21)2598-3247

[1]Partially supported by FONDECyT Grant 1050747 and Millennium Science Nucleus "Complex Engineering Systems", Chile and CNPq under PROSUL project Proc. 490333/2004-4, Brazil.

[2]Partially supported by UBACyT Grants X184 and X212, PICT ANPCyT Grant 11-09112 and PID Conicet Grant 644/98, Argentina and CNPq under PROSUL project Proc. 490333/2004-4, Brazil.

[3]Partially supported by UBACyT grant X184, Argentina.

[4]Partially supported by the Conselho Nacional de Desenvolvimento Científico e Tecnológico, CNPq, and Fundação de Amparo à Pesquisa do Estado do Rio de Janeiro, FAPERJ, Brasil.

ABSTRACT

A clique-transversal of a graph G is a subset of vertices intersecting all the cliques of G. It is NP-hard to determine the minimum cardinality τ_c of a clique-transversal of G. In this work, first we propose an algorithm for determining this parameter for a general graph, which runs in polynomial time, for fixed τ_c. This algorithm is employed for finding the minimum cardinality clique-transversal of $\overline{3K_2}$-free circular-arc graphs in $O(n^4)$ time. Further we describe an algorithm for determining τ_c of a Helly circular-arc graph in $O(n)$ time. This represents an improvement over an existing algorithm by Guruswami and Pandu Rangan which requires $O(n^2)$ time. Finally, the last proposed algorithm is modified, so as to solve the weighted version of the corresponding problem, in $O(n^2)$ time.

Keywords: algorithms, circular-arc graphs, clique-transversals, Helly circular-arc graphs, $\overline{3K_2}$-free circular-arc graphs.

1 Introduction

The aim of this work is to describe algorithms for finding the clique-transversal number for general graphs and for subclasses of circular-arc graphs: Helly circular-arc graphs and $\overline{3K_2}$-free circular-arc graphs.

Clique transversals have been studied since the paper by Tuza (1990). See also the early paper by Payan (1979). The first NP-hardness result for clique-transversals is by Erdös, Gallai and Tuza (1992). The following are some classes of graphs admitting polynomial time algorithms for the problem of determining a minimum clique-transversal: strongly chordal graphs (Chang, Farber and Tuza (1993), Chang, Chen, Chang and Yan (1996), Guruswami and Pandu Rangan (2000)); chordal graphs with bounded clique size (Guruswami and Pandu Rangan (2000)); k-trees with bounded k (Chang, Chen, Chang and Yan (1996)); dually chordal graphs (Brandstädt, Chepoi and Dragan (1997)); comparability graphs (Balachandran, Nagavamsi and Pandu Rangan (1996)); balanced graphs (Bonomo, Durán, Lin and Szwarcfiter (2005), (Dahlhaus, Manuel and Miller (1998)); distance hereditary graphs (Lee, Chang and Sheu (2002)); short-chorded graphs with no 3-fans nor 4-wheels (Durán, Lin and Szwarcfiter (2002)); Helly circular-arc graphs (Guruswami and Pandu Rangan (2000)).

Let G be an undirected connected graph, $V(G)$ and $E(G)$ its vertex and edge sets, respectively, $|V(G)| = n$ and $|E(G)| = m$. For $v \in V(G)$, denote by $N(v)$ the set of neighbors of v, and $N[v] = N(v) \cup \{v\}$. Write $\overline{N}(v) = V(G) \setminus N[v]$. Say that v is *universal* when $N[v] = V(G)$. A *complete set* of G is a set of pairwise adjacent vertices. A *clique* is a maximal complete set. A *dominating set* of G is a set $W \subseteq V(G)$ such that every vertex outside W is adjacent to some vertex of W. Let $\mathcal{V}$ a family of subsets of $V(G)$, and $W \subseteq V(G)$. Say that W is a *transversal* of $\mathcal{V}$ when W intersects each set of $\mathcal{V}$. A transversal of the set of cliques of G is called a *clique-transversal* of G.

We employ the following notation.

- $\tau_c(G)$, minimum cardinality of a clique-transversal of G, *clique-transversal number*
- $\tilde{\tau}_c(G)$, minimum weight of a clique-transversal of G,
- $\gamma(G)$, minimum cardinality of a dominating set of G, *domination number*
- $\tilde{\gamma}(G)$, minimum weight of a dominating set of G

A *circular-arc (CA) model* for G is a pair $(C, \mathcal{A})$, where C is a circle and $\mathcal{A}$ is a collection of arcs of C, such that each arc $A_i \in \mathcal{A}$ corresponds to a vertex $v_i \in V(G)$, and A_i, A_j intersect precisely when v_i, v_j are adjacent, $i \neq j$. A *circular-arc (CA) graph* is one admitting a CA model. When traversing the circle C, we will always choose the clockwise direction. If s, t are points of C, write (s, t) to mean the arc of C defined by traversing the circle from s to t. Call s, t the *extremes* of (s, t), while s is the *start* and t the *end* of the arc. For $A_i \in \mathcal{A}$, write $A_i = (s_i, t_i)$. Without loss of generality, all arcs of C are considered as open arcs, no two extremes of distinct arcs of $\mathcal{A}$ coincide and no single arc entirely covers C.

A *Helly circular-arc (HCA) graph* G is a CA graph admitting a CA model whose arcs satisfy the Helly property. That is, every pairwise intersecting subfamily of arcs of $\mathcal{A}$ contains a common point. Such a model is called a *Helly circular-arc (HCA) model* for G. Gavril (1974) has characterized HCA graphs as exactly those admitting a clique matrix having the circular $1'$s property for columns. This characterization leads to an algorithm for recognizing HCA graphs, which builds an HCA model in $O(n^3)$ time if that model exists.

A $\overline{3K_2}$-free circular-arc graph is a circular-arc graph which does not contain the graph of Figure 1 as an induced subgraph.

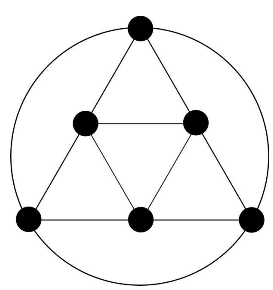

Figure 1: Graph $\overline{3K_2}$.

Helly circular-arc graphs form an important class of circular-arc graphs. Some properties of interval graphs are captured more closely by Helly circular-arc graphs than by other classes of circular-arc graphs. On the other hand, $\overline{3K_2}$-free circular-arc graphs contain Helly circular-arc graphs and their cliques preserve some of the properties of the latter class.

In this work, we propose an algorithm for determining the minimum cardinality clique-transversal of a general graph, which runs in polynomial time, whenever the clique-transversal number is fixed. We also describe efficient algorithms for finding clique-transversals in certain subclasses of circular-arc graphs. The considered classes are Helly circular-arc graphs and $\overline{3K_2}$-free circular-arc graphs. For Helly circular-arc graphs, we propose algorithms for the cardinality and weighted version of this problem. For $\overline{3K_2}$-free circular-arc graphs, we describe an algorithm for minimum cardinality clique-transversal. The complexity of the proposed algorithm for the cardinality problem in HCA graphs is $O(n)$. This represents an improvement over the existing algorithm by Guruswami and Pandu Rangan (2000), whose complexity is $O(n^2)$. As usual for many algorithms on circular-arc graphs, we assume that the graph is given by its circular-arc model, with the extremes of the arcs circularly sorted. If they are not sorted we would need to add an extra $O(n \log n)$ time for the sorting. All the mentioned algorithms for HCA graphs suppose that an HCA model is given.

2 Clique-transversals in general graphs

In the sequel, we consider the question of finding the clique-transversal number for an arbitrary graph.

The theorem below describes conditions for an arbitrary graph G to have clique-transversal number at most k.

Theorem 1 *Let G be a graph and $k \geq 1$. Then $\tau_c(G) \leq k$ if and only if G has k vertices $v_1, \ldots, v_k$, such that the family of subsets $\overline{N}(v_1), \ldots, \overline{N}(v_k), \subseteq V(G)$ admits no transversal formed by a complete set of $V(G)$.*

Proof: Assume $\tau_c(G) \leq k$ and let $\{v_1, \ldots, v_k\}$ be a clique-transversal of G. By contrary, suppose that $\overline{N}(v_1), \ldots, \overline{N}(v_k)$ has a transversal W, which is a complete set of G. Because W is a complete set, W is contained in some clique M of G. Since W is a transversal of $\overline{N}(v_1), \ldots, \overline{N}(v_k)$, at least one vertex of M is not adjacent to v_i, $1 \leq i \leq k$. Consequently, $M \cap \{v_1, \ldots, v_k\} = \emptyset$. The latter contradicts $\{v_1, \ldots, v_k\}$ to be a clique-transversal of G. Consequently, $\overline{N}(v_1), \ldots, \overline{N}(v_k)$ has no transversal formed by a complete set of G.

Conversely, by hypothesis $\tau_c(G) > k$. By contradiction, assume that G has k vertices $\{v_1, \ldots, v_k\}$ such that $\overline{N}(v_1), \ldots, \overline{N}(v_k)$ has no transversal formed by a complete set of G. Because $\tau_c(G) > k$, there exists some clique M such that $M \cap \{v_1, \ldots, v_k\} = \emptyset$. Then M contains a vertex $w \in \overline{N}(v_i)$, for each i. The collection of such vertices w form a complete set W with cardinality at most k, and which is a transversal of $\overline{N}(v_1), \ldots, \overline{N}(v_k)$, contrary to the assumption. Consequently, no such vertices $v_1, \ldots, v_k$ may exist. $\triangle$

The above theorem conducts to the following algorithm for determining whether the clique-transversal number of an arbitrary graph G is at most k, for a given k.

Algorithm 1 *DECIDING IF THE CLIQUE-TRANSVERSAL NUMBER OF A GENERAL GRAPH IS AT MOST k*

For each k-subset $V' = \{v_1, \ldots, v_k\} \subseteq V(G)$, consider all subsets $W \subseteq V(G) \setminus V'$, with $|W| \leq k$. For each pair V', W, verify (i) if W is a transversal of $\overline{N}(v_1), \ldots, \overline{N}(v_k)$ and (ii) if W is a complete set of G. If, for some V', (i) or (ii) fails for all subsets $W \subseteq V(G) \setminus V'$, then V' is a clique-transversal of G, $\tau_c(G) \leq k$ and stop. Otherwise, $\tau_c(G) > k$.

The complexity of the above algorithm can be determined as follows. There are $O(kn^{2k})$ pairs of subsets V', W to be considered. For each V', we can restrict to consider only those subsets W which are already a transversal of $\overline{N}(v_1), \ldots, \overline{N}(v_k)$, that is, which satisfy (i). To verify (ii), we require $O(k^2)$ time. Consequently, the overall time complexity is $O(k^3n^{2k})$, with $O(m+nk)$ space.

By applying $\tau_c(G)$ times the above algorithm, we can compute the value of $\tau_c(G)$. The complexity is therefore a polynomial in n, for fixed $\tau_c(G)$.

3 Intersection segments

Let G be a graph admitting a CA model $(C, \mathcal{A})$. For $A \in \mathcal{A}$, denote by $V(A)$ the vertex of G corresponding to A. Similarly, for $\mathcal{A}' \subseteq \mathcal{A}$, $V(\mathcal{A}') = \cup_{A \in \mathcal{A}'} V(A)$. If $V(A)$ is a universal vertex then A is a *universal arc*. If an arc $A \in \mathcal{A}$ contains some point $p \in C$ then say that A is an arc of p. Denote by $\mathcal{A}(p)$ the collection of arcs of p. Clearly, $V(\mathcal{A}(p))$ is complete set of G. For $p, p' \in C$ say that p *(properly) dominates* p' when $\mathcal{A}(p)$ (properly) contains $\mathcal{A}(p')$. When $\mathcal{A}(p) = \mathcal{A}(p')$ then p, p' are *equivalent*. Say that $p \in C$ is *complete point* when no point of C properly dominates p. In addition when $V(\mathcal{A}(p))$ is a clique of G then p is a *clique point* of C. Such a clique is called a *Helly clique*. Clearly, G might contain cliques that are no Helly. However, if $(C, \mathcal{A})$ is a Helly model then all its cliques are Helly. In this case, there is a one-to-one correspondence between cliques of G and non equivalent clique points of C. On the other hand, any non Helly clique contains at least three vertices. Furthermore, among the arcs of $\mathcal{A}$ corresponding to the vertices of a non Helly clique there exist always three of them which together cover the entire circle.

We describe a method for finding sets of complete points of a CA graph. These sets will be employed in the algorithms proposed in the later sections. The following concepts are central for our methods.

Denote $\mathcal{A} = \{A_1, \ldots, A_n\}$ and $A_i = (s_i, t_i)$, $1 \leq i \leq n$. A *segment* is an arc of C formed by two consecutive extremes of the arcs of $\mathcal{A}$, when traversing C. Clearly, there are $2n$ segments, which exactly cover C, except for their extreme points. Also, each arc of $\mathcal{A}$ corresponds to a sequence of consecutive segments. All points belonging to a same segment are equivalent. An *intersection segment* is a segment of the type (s_i, t_j), that is, its start point is the start point of some arc arc $A_i \in \mathcal{A}$, while its end point is the end point of an arc $A_j \in \mathcal{A}$. Write $I_i = (s_i, t_j)$. A point $p_i \in I_i$ is called an *intersection point*. There are at most n intersection segments.

In order to relate intersection points to complete points, we employ the following additional notation. An intersection segment $I_i = (s_i, t_j)$ is *simple* when $A_i \cup A_j \neq C$, and *universal* otherwise. That is, I_i is universal when A_i and A_j cover the entire circle. A point belonging to a simple segment is a *simple point*, whereas one inside a universal segment is a *universal point*.

Next, we consider some special subsets of points of C which are of interest. Define the following four subsets. A *complete (simple) (universal) (clique) point representation of* C is a maximal set of complete (simple) (universal) (clique) non equivalent points of C. Represent these sets by P, S, U, Q, respectively. We describe how to construct them.

Let $P', P'' \subseteq C$ be two subsets of points of C. Then P', P'' are *isomorphic* when there exists a bijection f between these sets such that p' and $f(p')$ are equivalent, for all $p' \in P'$. Clearly, any two complete (intersection) (simple) (universal) (clique) point representations are isomorphic. That is, P, S, U, Q are all unique, up to isomorphism. Consequently, we can write $P = S \cup U'$, where $U' \subseteq U$. Also, $Q \subseteq P$, with $Q = P$ precisely when $(C, \mathcal{A})$ is a Helly model. Clearly, Q corresponds to the set of Helly cliques of G. Moreover, the Helly cliques can be further bipartitioned, as follows. Let M_i be a Helly clique of G and p_i the clique point of Q corresponding to M_i. Then M_i is a *simple clique* or *universal clique*, according whether p_i is a simple or universal point, respectively.

The following algorithm proposed in (Durán, Lin, Mera and Szwarcfiter (2005)) constructs a complete point representation P of C, given a CA model $(C, \mathcal{A})$ for a graph G. In fact, the algorithm constructs explicitly the simple point representation S and then finds $U' \subseteq U$, such that $P = S \cup U'$. The algorithm is divided into two steps. Step 1 constructs S and a set $U'' \supseteq U$, which contains U and possibly some additional equivalent points. Step 2 determines U' by including in it one universal point (the one with lowest index), for each collection of equivalent complete points. The algorithm employs a list L to contain this collection.

Algorithm 2 *CONSTRUCTING A COMPLETE POINT REPRESENTATION OF A CA GRAPH*

STEP 1: Identify the segments of C. Define $S = U'' = \emptyset$. For each segment (x, y) of C, if x is the start of some arc $A_i \in \mathcal{A}$ and y the end of $A_j \in \mathcal{A}$ then let p_i be a point of (x, y) and perform the following additional test: if $A_i \cup A_j \neq C$, include p_i in S, otherwise include p_i in U''.

STEP 2: Define $U' = \emptyset$. For each universal point $p_i \in U''$, let $I_i = (s_i, t_j)$ be its corresponding universal segment. For each $p_i \in U''$, apply the following procedure. Compute $\mathcal{A}(p_i)$. Define $L = \{i\}$. Traverse the arc $(s_j, t_i) \subseteq C$, segment by segment, in the order as they appear. In case of an intersection segment $(s_k, t_l) \subseteq (s_j, t_i)$, choose a point $p_k \in (s_k, t_l)$, compute $\mathcal{A}(p_k)$, and if $\mathcal{A}(p_i) = \mathcal{A}(p_k)$ then include k in L. After all segments contained in (s_j, t_i) have been traversed then include p_r in U' precisely in the case where p_i is not properly dominated by any p_k, and $r = min\{k \in L\}$. At the end, $P = S \cup U'$.

Algorithm 2 constructs S and U'' in $O(n)$ time and $U' \subseteq U''$ in $O(n^2)$ time. Consequently, we require $O(n^2)$ time for constructing P. For determining U, possibly we need to eliminate equivalent points from the subset U' constructed in Step 1. It can be easily performed in overall $O(n^2)$ time.

Finally, consider the determination of the clique point representation Q of C. To obtain $Q \subseteq P$, we need to remove from P those points $p \in P$, such that $V(\mathcal{A}(p))$ is not a clique. With this purpose, apply the following algorithm, proposed in (Durán, Lin, Mera and Szwarcfiter (2005)). Given P the algorithm constructs Q in $O(m)$ time.

Algorithm 3 *CONSTRUCTING A CLIQUE POINT REPRESENTATION OF A CA GRAPH*

Define $Q := P$. For each complete point $p \in P$, perform the following operations. Denote by (s_i, t_j) the intersection segment corresponding to p. Define $s_o := s_j$. Traverse the arc (t_j, t_i), identifying the extreme points q of the arcs $A_k \in \mathcal{A}$, such that q is the first extreme of A_k, in the traversal. For each such extreme q, do the following: if $q = s_k$ and $t_k \in (s_0, s_i)$ then $Q := Q \setminus \{p\}$ and terminate the iteration corresponding to p (p is not a clique point*); if $q = t_k$ and $s_k \in (s_0, s_i)$ then assign $s_0 := s_k$. At the end, Q is the required clique point representation.*

4 Clique-transversals in HCA graphs

In this section, we describe a method for finding a minimum clique transversal of an HCA, for the weighted and cardinality versions of the problem.

Let G be an HCA graph and $(C, \mathcal{A})$ an HCA model for it. Let M_i be a clique of G and p_i the clique point of C corresponding to it. Denote by G' the graph obtained from G, by adding a new vertex w_i, for each clique M_i of G, making w_i adjacent exactly to the vertices of M_i. Clearly, G' is also an HCA graph, as an HCA model for it can be obtained from $(C, \mathcal{A})$, by including in $\mathcal{A}$ a new arc A'_i for each clique M_i of G. Each A'_i includes p_i and is contained in the intersection of the arcs of p_i, but containing none of the extremes of this intersection. Note that each A'_i creates a new intersection segment in G'. Call G' the *simplicial augmentation* of G.

The following theorem relates clique-transversals of G and dominating sets of G'.

Theorem 2 *Let G be an HCA graph, G' its simplicial augmentation and $W \subseteq V(G)$. Then W is a clique-transversal of G if and only if W is a dominating set of G'.*

Proof: Let (C, A) be an HCA model for G. Denote by Q the clique point representation of C. Let $\mathcal{A}_Q$ be a family of $|Q|$ arcs of C, each one containing exactly one clique point of Q and no extremes of arcs of $\mathcal{A}$. It follows that $(C, \mathcal{A} \cup \mathcal{A}_Q)$ is an HCA model for G'. Let $\mathcal{A}_W \subseteq \mathcal{A}$ be the set of arcs of $\mathcal{A}$ corresponding to the vertices of W.

Assume that W is a clique-transversal of G. Then the vertices of W meet each clique M_i of G. That is, each clique point $p_i \in Q$ is covered by some arc of $\mathcal{A}_W$. On the other hand, each arc of $\mathcal{A} \cup \mathcal{A}_Q$ contains some clique point $p_i \in Q$. Hence, the collection $\mathcal{A}_W$ intersects all arcs of $\mathcal{A} \cup \mathcal{A}_Q$, meaning that W is a dominating set of G'.

Conversely, suppose that $W \subseteq V(G)$ is a dominating set of G'. Then each arc of $\mathcal{A} \cup \mathcal{A}_Q$ intersects some arc of $\mathcal{A}_W$. In particular, for any clique point $p_i \in Q$, the arc $A'_i \in \mathcal{A}_Q$ which contains $p_i \in Q$ intersects some arc $A_w \in \mathcal{A}_W$. We know that any arc of $\mathcal{A} \cup \mathcal{A}_Q$ which intersects A'_i must contain A'_i. Consequently, A_w contains p_i, meaning that W is indeed a clique-transversal of G. $\triangle$

We handle separately CA graphs having two arcs covering the entire circle.

Theorem 3 *Let $(C, \mathcal{A})$ be a CA model of a graph G. If there are two arcs $A_1, A_2 \in \mathcal{A}$ which entirely cover C then $\tau_c(G) \leq 2$.*

Proof: Let M be any clique of G and $A(M) = \{A_i \in \mathcal{A} \ / \ V(A_i) \in M\}$. If A_1 or A_2 contains some arc $A_i \in A(M)$ different from A_1 and A_2, then $V(A_1)$ or $V(A_2)$ belongs to M. If A_1 and A_2 do not contain any arc of $A(M)$ different from A_1 and A_2, then $\forall A_i \in A(M)$, $A_i \cap A_1 \neq \emptyset$ and $A_i \cap A_2 \neq \emptyset$, because A_1 and A_2 entirely cover C. In this case, $V(A_1)$ and $V(A_2)$ are vertices of M. Consequently, $V(A_1)$and $V(A_2)$ form a clique transversal of G.$\triangle$

In the sequel, we apply Theorems 2 and 3 for finding the clique-transversal number of an HCA graph G, given by its HCA model $(C, \mathcal{A})$. The following algorithm computes $\tau_c(G)$.

Algorithm 4 *CLIQUE-TRANSVERSAL NUMBER OF AN HCA GRAPH*

Start by verifying if $\mathcal{A}$ contains a universal arc. If affirmative, $\tau_c(G) = 1$. Otherwise, construct the simple and universal point representations S and U of G, respectively. If $U \neq \emptyset$ then $\tau_c(G) = 2$. Otherwise, find the collection of arcs $\mathcal{A}_S$ and the HCA model $(C, \mathcal{A} \cup \mathcal{A}_S)$ of the simplicial augmentation G' of G. Then $\tau_c(G) = \gamma(G')$.

Clearly, $\tau_c(G) = 1$ precisely when G has a universal arc. Otherwise and when $U \neq \emptyset$ there are two arcs which cover C, meaning that $\tau_c(G) = 2$. Otherwise, $U = \emptyset$ implies that the clique point representation Q equals S. This means that $Q = P = S$. Consequently, the construction of Q reduces to that of S, which can be done in $O(n)$ time, running Step 1 of Algorithm 2. The construction of $\mathcal{A}_S$ and of the HCA model of G' can also be done in linear time. In order to compute $\gamma(G')$, apply the algorithm by Hsu and Tsai (1991) , which runs in $O(n)$ time. Consequently, the overall complexity is $O(n)$.

The above algorithm can be modified for the weighted problem. Let G be a graph given by its HCA model $(C, \mathcal{A})$, and where there is a non negative weight assigned to each of its vertices.

Algorithm 5 *MINIMUM WEIGHT OF A CLIQUE-TRANSVERSAL OF AN HCA GRAPH*

Construct the clique point representation Q of C, and the family of arcs $\mathcal{A}_Q$. Find the HCA model $(C, \mathcal{A} \cup \mathcal{A}_Q)$ of the simplicial augmentation G' of G. Define the weights of the vertices of G', as follows. For $v \in V(G)$, the weight of v in G' is the same as its weight in G, while the weight of a vertex $v \in V(G') \setminus V(G)$ is infinite. Then $\tilde{\tau}_c(G) = \tilde{\gamma}(G')$.

The infinite weights assure that the minimum dominating set of G' is formed solely by vertices of G. By Theorem 2, the algorithm is correct.

The construction of Q requires $O(n^2)$ time, by Algorithms 2 and 3. The determination of $\tilde{\gamma}(G')$ can be done in $O(n+m)$ time, applying the algorithm by Chang (1998). The remaining operations can de implemented in $O(n)$ time. Therefore the algorithm terminates within $O(n^2)$ time.

5 Clique-transversals in $\overline{3K_2}$-free CA graphs

Finally, we consider the problem of finding $\tau_c(G)$ for a $\overline{3K_2}$-free CA graph G.

Theorem 4 *Let G be a graph which is not HCA, and contains no $\overline{3K_2}$ as induced subgraph. Then $\tau_c(G) \leq 3$.*

Proof: Let $(C, \mathcal{A})$ be a CA model for G. As G is not an HCA graph, there are three arcs $A_1, A_2, A_3 \in \mathcal{A}$, which entirely cover C. If two of these arcs cover C then it holds that $\tau_c(G) \leq 3$ by Theorem 5. So, we can assume that A_1, A_2, A_3 do not have a common point. Suppose there is some clique M which is not covered by the subset of vertices $V(A_1)$, $V(A_2)$ and $V(A_3)$. That is, there exists an arc A_1', corresponding to some vertex of M, such that $A_1' \cap A_1 = \emptyset$, but $A_1' \cap A_2$, $A_1' \cap A_3 \neq \emptyset$. The latter can be justified as follows: $A_1' \cap A_1 = \emptyset$, otherwise $V(A_1) \in M$, while $A_1' \cap A_2$, $A_1' \cap A_3 \neq \emptyset$, because A_1, A_2, A_3 cover the entire circle and A_1' can not be strictly contained in A_2 nor A_3. Similarly, there exist arcs A_2' and A_3' also corresponding to vertices of M, satisfying $A_2' \cap A_2 = A_3' \cap A_3 = \emptyset$, but $A_2' \cap A_1$, $A_2' \cap A_3$, $A_3' \cap A_1$, $A_3' \cap A_2 \neq \emptyset$. In this situation, the subset of vertices corresponding to the arcs $\{A_1, A_2, A_3, A_1', A_2', A_3'\}$ induces a $\overline{3K_2}$ in G, which contradicts the hypothesis. Consequently, M is covered by $V(A_1)$, $V(A_2)$ and $V(A_3)$. Therefore $V(A_1)$, $V(A_2)$ and $V(A_3)$ form a clique transversal of G.$\triangle$

The above proof also implies that if G is a $\overline{3K_2}$-free CA graph and $\tau_c(G) > 3$ then every CA model for G is in fact an HCA model.

Theorem 4 leads to the following algorithm for determining the clique-transversal number of a $\overline{3K_2}$-free CA graph G, with a given CA model $(C, \mathcal{A})$.

Algorithm 6 *CLIQUE-TRANSVERSAL NUMBER OF A $\overline{3K_2}$-FREE CA GRAPH*

Start by verifying if G contains a universal vertex. If affirmative then $\tau_c(G) = 1$. Otherwise, apply Algorithm 1 with $k = 2$, to verify if $\tau_c(G) \leq 2$. If affirmative, $\tau_c(G) = 2$. Otherwise, check if there are three arcs $A_1, A_2, A_3 \in \mathcal{A}$ which cover C. If negative, then $(C, \mathcal{A})$ is an HCA model and determine $\tau_c(G)$ by applying Algorithm 4. When A_1, A_2, A_3 cover C, verify if there are other three arcs $A_4, A_5, A_6 \in \mathcal{A}$ covering C (in this case, $A_1, \ldots, A_6$ form a $\overline{3K_2}$ in G). In the affirmative case, the algorithm exhibits such a subgraph. Otherwise, $\tau_c(G) = 3$.

This algorithm is robust, in the sense that either it determines the clique transversal number of the graph, or it exhibits a forbidden $\overline{3K_2}$ induced subgraph.

As for the complexity, the dominating step is that of applying Algorithm 1 with $k = 2$. Consequently, the algorithm terminates in $O(n^4)$ time.

6 Conclusions

The table below summarizes the problems that have been considered in this paper, together with the complexities of the corresponding proposed algorithms.

Problem	Graph Class	Version	Proposed alg.	Previous alg.
Clique-transversal number	HCA	cardinality	$O(n)$	$O(n^2)$
		weighted	$O(n^2)$	-
	$\overline{3K_2}$-free CA	cardinality	$O(n^4)$	-
		weighted	?	-
	general	cardinality	$O(\tau_c^4(G).n^{2\tau_c(G)})$	-

In all cases, the algorithms determine the cardinality or the weight of the corresponding minimum clique-transversal set. There is no difficulty to modify them so as to compute the actual minimum or maximum sets.

It remains open the complexity of determining the clique-transversal number of general CA graphs.

References

[1] Balachandran, V., Nagavamsi, P. and Pandu Rangan, C (1996). "Clique Transversal and Clique Independence on Comparability Graphs", *Information Processing Letters* 58, 181-184.

[2] Bonomo, F., Durán, G., Lin, M. C., and Szwarcfiter, J. L. (2005). "On Balanced Graphs", *Mathematical Programming B*, to appear.

[3] Brandstädt, A., Chepoi V. D., and Dragan, F. F. (1997). "Clique r-Domination and Clique r-Packing Problems on Dually Chordal Graphs", *SIAM Journal on Discrete Mathematics* 10, 109-127.

[4] Chang, G., J., Farber, M., Tuza, Zs. (1993). "Algorithmic Aspects of Neighbourhood Numbers", *SIAM Journal on Discrete Mathematics* 6, 24-29.

[5] Chang, M., S. (1998). "Efficient Algorithms for the Domination Problems on Interval and Circular-Arc Graphs", *SIAM Journal on Computing* 27, 1671-1694.

[6] Chang, M. S., Chen, Y. H., Chang, G. J., and Yan, J. H. (1996). "Algorithmic Aspects of the Generalized Clique-Transversal Problem on Chordal Graphs", *Discrete Applied Mathematics* 66, 189-203.

[7] Dahlhaus, E., Manuel, P. D., and Miller, M. (1998). "Maximum h-Colourable Subgraph Problem in Balanced Graphs", *Information Processing Letters 65*, 301-303.

[8] Durán, G., Lin, M. L., Mera, S., and Szwarcfiter, J. L. (2005). "Algorithms for Clique-Independent Sets on Subclasses of Circular-Arc Graphs", submitted.

[9] Durán, G., Lin, M. L., and Szwarcfiter, J. L. (2002). "On Clique-Transversals and Clique-Independent Sets", *Annals of Operations Research* 116, 71-77.

[10] Erdös, P., Gallai, T., and Tuza, Zs. (1992). "Covering the Cliques of a Graph with Vertices", *Discrete Mathematics*, 279-289.

[11] Gavril, F. (1974). "Algorithms on Circular-Arc Graphs", *Networks 4*, 357-369.

[12] Guruswami, V., and Pandu Rangan, C. (2000). "Algorithmic Aspects of Clique Transversals and Clique Independent Sets", *Discrete Applied Mathematics* 100, 183-202.

[13] Hsu, W. L., and Tsai, K. H. (1991). "Linear Time Algorithms on Circular-Arc Graphs", *Information Processing Letters* 40, 123-129.

[14] Lee, C. M., Chang, M. S., and Sheu, S. C. (2002). "The Clique Transversal and Clique Independence of Distance Hereditary Graphs", In *Proceedings of the 19th Workshop on Combinatorial Mathematics and Computation Theory*, 64-69, Taiwan.

[15] Payan, C. (1979). "Remarks on Cliques and Dominating Sets in Graphs", *Ars Combinatoria*, 181-189.

[16] Tuza, Zs. (1990). "Covering all Cliques of a Graph", *Discrete Mathematics* 86, 117-126.