LIF

Laboratoire d'Informatique Fondamentale
de Marseille

Unité Mixte de Recherche 6166
CNRS – Université de Provence – Université de la Méditerranée

Mutual exclusion scheduling with interval graphs and related classes

Frédéric Gardi

Rapport/Report 12-2003

May 2003

Mutual exclusion scheduling with interval graphs and related classes

Frédéric Gardi

LIF – Laboratoire d'Informatique Fondamentale de Marseille
UMR 6166
CNRS – Université de Provence – Université de la Méditerranée

Parc Scientifique et Technologique de Luminy,
163, avenue de Luminy - Case 901,
F-13288 Marseille Cedex 9, France.

gardi@lidil.univ-mrs.fr

Abstract/Résumé

In this note, the mutual exclusion scheduling problem is considered. Given an undirected graph G and an integer k, the problem is to find a minimum coloring of G such that each color is used at most k times. The cardinality of such a coloring is denoted with $\chi(G, k)$. For intervals graphs, the problem is known to be $\mathcal{NP}$-complete even for fixed $k \geq 4$. We prove that if an interval graph or more generally a circular-arc graph G with n vertices admits a coloring such that each colour is used at least k times, then $\chi(G, k)$ equals the lower bound $\lceil n/k \rceil$. The proof yields a linear-time algorithm to solve the problem in this case which finds applications in schedules planning. Then, the assertion is extended to the class of triangulated graphs for $k \leq 3$ and disproved for bounded tolerance graphs and co-comparability graphs.
Keywords: scheduling, colorings, interval graphs, classes of graphs.

Dans cette note, un problème d'ordonnancement avec exclusion mutuelle est abordé. Etant donné un graphe non-orienté G et un entier k, le problème consiste à trouver une coloration de G tel que chaque couleur apparaisse au plus k fois. La cardinalité d'une telle coloration est notée $\chi(G, k)$. Pour les graphes d'intervalles, le problème est $\mathcal{NP}$-complet même pour une valeur fixée $k \geq 4$. Nous prouvons que si un graphe d'intervalles ou plus généralement un graphe d'arc circulaires G à n sommets admet une coloration tel que chaque couleur est utilisée au moins k fois, alors $\chi(G, k)$ égale la borne inférieure $\lceil n/k \rceil$. La preuve fournit un algorithme linéaire pour résoudre le problème dans ce cas qui possède des applications dans la planification d'horaires de travail. Ensuite, ce résultat est étendu à la classe des graphes triangulés pour $k \leq 3$ et réfuté pour les graphes de tolérance bornée et les compléments des graphes de comparabilité. **Mots-clés :** ordonnancement, colorations, graphes d'intervalles, classes de graphes.

Relecteurs/Reviewers: Michel Van Canegehem, Alain Colmerauer.

1 Introduction

Definition of the problem. The following problem arises in scheduling theory: n unit-time jobs must be complete on k processors in a minimum time with the constraint that some jobs cannot be executed at the same time because they share a same ressource. Many variants of this problem have been considered in combinatorial optimization and operations research litterature (see [19, 5, 4, 16, 1] for different applications). Such scheduling problems can be alternatively formulated in graph-theoretic terms. By creating an undirected graph $G = (V, E)$ with a vertex for each of the n jobs and an edge between each pair of conflicting jobs, we can see that a schedule of minimum length corresponds to a partition of V into a minimum number of independent sets of size at most k. In this way, B.S. Baker and E.G. Coffman [1] called *Mutual Exclusion Scheduling* (MES) the following graph-theoretical problem: given an undirected graph G and a natural number k, find a minimum coloring of G such that each colour is used at most k times. The cardinality of this a coloring shall be denoted by $\chi(G, k)$. A trivial lower bound for $\chi(G, k)$ is given by $\max(\chi(G), \lceil n/k \rceil)$, where $\chi(G)$ is the chromatic number of the graph G.

Since coloring a graph is a $\mathcal{NP}$-complete problem [18], MES is $\mathcal{NP}$-complete too. Then, MES was studied for classes of graphs for which the coloring problem is polynomial, notably the perfect graphs (see [11] for an introduction to the world of perfect graphs). Unfortunately, MES remains $\mathcal{NP}$-complete for many basic classes of perfect graphs: bipartite graphs, cographs, interval and triangulated graphs [5], permutation and comparability graphs [17]. At our knowledge, MES is proved to be polynomial-time solvable only for forests and trees [1], split graphs [20], complements of interval graphs [5] and proper interval graphs [7].

Results and applications. Our interest to the MES problem for interval graphs comes from a *working schedules planning* problem (WSP), which has actually inspired this research. This problem has the following definition. Let $\{T_i\}_{i=1,\dots,n}$ be a set of tasks having each one a starting date l_i and an ending date r_i. The reglementation imposes that an employee cannot execute more than k tasks. Given that the tasks allocated to an employee must not overlap, find a planning requiring the minimum number of employees. Since the tasks are some intervals of the real line, the WSP problem is equivalent to the MES problem for interval graphs. If the tasks are cyclic then the problem becomes equivalent to the MES for circular-arc graphs.

A undirected graph $G = (V, E)$ is an *interval graph* if to each vertex $v \in V$ an interval $I_v = [le(I_v), re(I_v)]$ on the real line can be associated such that each pair of distinct vertices $u, v \in V$ are adjacent if and only if $I_u \cap I_v \neq \emptyset$. The family $\{I_v\}_{v \in V}$ is an *interval representation* of G. The intersection graph obtained from collections of arcs on a circle is called *circular-arc graph.* A *circular-arc representation* $\{A_v\}_{v \in V}$ of an undirected graph G which fails to cover some point p on the circle will be topologically the same as an interval representation of G. Specificially, we can cut the circle at p and straighten it out a line, the arcs becoming intervals. It is easy to see therefore, that every interval graph is a circular-arc graph.

Interval graphs are colorable in linear time [11, 14, 22] whereas coloring circular-arc graphs is an $\mathcal{NP}$-complete problem [8]. Nevertheless, MES for interval graphs is $\mathcal{NP}$-complete even for fixed $k \geq 4$ (at our acquaintance the problem for $k = 3$ remains an open question) [5]. Moreover, the reduction given in [5] can be slightly modified to establish the $\mathcal{NP}$-completeness of MES for circular-arc graphs with fixed $k \geq 3$. In a first section, the following sufficient condition is given to obtain $\chi(G, k) = \lceil n/k \rceil$ for MES with interval and circular-arc graphs: there exists a coloring of the graph G such that each color is used at least k times. The constructive proof yields a linear-time algorithm to solve the MES problem in this case (given the graph and the coloring in input). One corollary of this result is: for an interval graph (or a circular-arc graph) G, if $\chi(G, k) = \lceil n/k \rceil$ then for all $k' < k$, we have $\chi(G, k') = \lceil n/k' \rceil$ too. Such a result finds applications in WSP of certain municipal bus drivers or air terminal personnels (schedules planning problems solved by the firm PROLOGIA–Groupe Air Liquide [2]). Indeed, the movements of buses or planes in these cases generated some packets of consecutive tasks inducing independent sets of size larger than k (for reasonnable values of k like $3, 4, 5$).

Extensions. Two common super-classes of interval graphs are the classes of triangulated graphs and complements of comparability graphs. *Triangulated graphs* (also called chordal graphs) are the graphs without a cycle of length four or more as an induced subgraph. A triangulated graph can be represented by an intersection graph of a family of subtrees of a tree. Thus, each interval graph is triangulated. The complement $\overline{G} = (V, F)$ of an interval graph G can be transitively oriented by setting $(u, v) \in \vec{F}$ if $re(I_u) < le(I_v)$ (we shall write $I_u \prec I_v$ if $re(I_u) < le(I_v)$). Transitively orientable (or partially orderable) graphs are also known as *comparability graphs*. Hence, each interval graph is the complement of a comparability graph, shortly a co-comparability graph. Interval graphs are exactly the graphs which are both triangulated and co-transitively orientable [10]. Another interesting class which contains interval graphs is the class of tolerance graphs. A graph $G = (V, E)$ is a *tolerance graph* if each vertex $v \in V$ can be assigned an interval I_v and a positive real number t_v referred to as its tolerance, such that for each pair of distinct vertices, $uv \in E$ if and only if $|I_u \cap I_v| \geq \min(t_u, t_v)$. The family $\{I_v\}_{v \in V}$ is a *tolerance representation* of G. If a graph has a tolerance representation such that the tolerance of each vertex $v \in V$ is smaller than the length of I_v, then this graph is called a *bounded tolerance graph*. Bounded tolerance graphs are co-comparability graphs [13].

Triangulated graphs are colorable in linear time [9] whereas coloring co-comparability or tolerance graphs takes $O(n^3)$ time [11]. The MES problem is $\mathcal{NP}$-complete for triangulated and co-comparability graphs even for fixed $k \geq 3$ [20, 17]. In a second section, we study the impact of the condition previously defined on MES with these super-classes of interval graphs. We extend the result of the first section to the triangulated graphs for $k = 2, 3$ (we conjecture that the property holds for any integer k). Finally, we give counter-examples for bounded tolerance graphs and also for co-comparability graphs with any integer k.

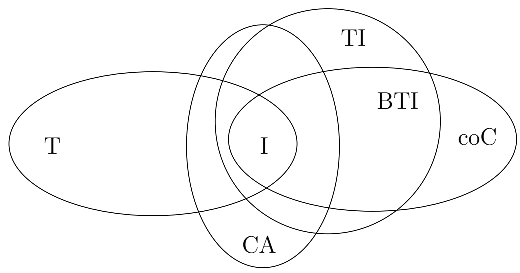

Fig. 1. Inclusion hierarchy of interval graphs (I), circular-arc graphs (CA), triangulated graphs (T), tolerance graphs (TI), bounded tolerance interval graphs (BTI) and co-comparability graphs (coC).

Fig. 1 summarizes the inclusion hierarchy of the classes of graphs previously defined. For more details on these graphs and their applications, the interested reader can consult [23, 11, 3, 12] for surveys. All graph-theoretical terms not defined here can be found in [3, 11].

2 A sufficient condition for MES with interval and circular-arc graphs

First we prove the validity of the sufficient condition for MES with interval graphs.

Proposition 1 *Let G be an interval graph with n vertices and k an integer. If G admits a coloring such each color is used at least k times, then $\chi(G, k) = \lceil n/k \rceil$. Moreover, a partition of G into $\lceil n/k \rceil$ independent sets can be computed in $O(n)$ time and $O(n)$ space given an ordered interval representation and the coloring in input.*

The proof is essentially based on the following lemma.

Lemma 1 *Let $S_1, \ldots, S_t$ be t independent sets of G satisfying the following conditions:*

. $t \in \{1, \ldots, k\}$,

. *for* $i = 1, \ldots, t$, $|S_i| = k + r_i$ *with* $r_i \in \{1, \ldots, k-1\}$,

. $r = \sum_{i=1}^{t} r_i \geq k$.

Then there exists an independent set S^* of size r such that for all $i = 1, \ldots, t$, r_i intervals of S^* belong to S_i. In other words, $S_1, \ldots, S_t$ admits an optimal partition into t independent sets of size k and one independent set of size r.

Proof. An algorithm having in input an interval representation of $S_1, \ldots, S_t$ is proposed for the construction of S^*. The intervals of each independent set are supposed to be ordered according to the relation $\prec$. The *rank* of an interval in its independent set is its number in this order. In this way, $I_{i,j}$ denotes the interval of rank j in the independent set i. At each step, the algorithm selects one interval among the t independent sets, removes it from its independent set and includes it in S^*. After k steps, the independent set S^* is returned in output. The selection of I_j^* (*ie.* the interval of rank j in S^*) is done as follows: choose the interval having the smallest right endpoint among the intervals of rank j, which belong to independent sets of size still larger than k (see Fig. 2). The complete procedure is detailed below.

Algorithm ExtractStable;
Input: k an integer,
  $S_1, \ldots, S_t$ some independent sets satisfying the conditions of Lemma 6;
Output: $S^* = \{I_1^*, \ldots, I_k^*\}$ an independent set;
begin;
  $S^* \leftarrow \emptyset$;
  for $j \leftarrow 1, \ldots, r$ **do**
    $F \leftarrow \emptyset$;
    for $i \leftarrow 1, \ldots, t$ **do**
      if $|S_i| > k$ **then**
        $F \leftarrow F \cup \{I_{i,j}\}$;
    let I_j^* be the interval having the smallest right endpoint in F;
    remove I_j^* from its independent set and add it to S^*;
  return S^*;
end;

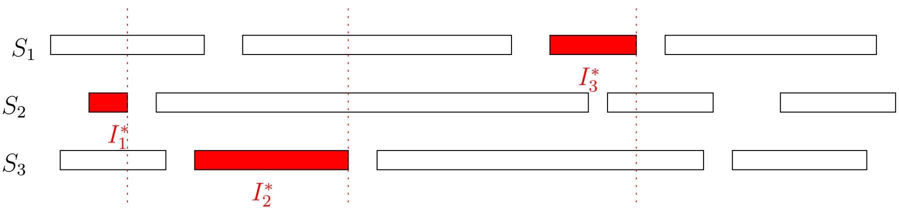

Fig. 2. An example of the execution of the algorithm with $k = 3$.

To conclude, the correctness of the algorithm is established. At each step of the algorithm, an interval is selected (every input independent set has at least k intervals). Therefore, S^* contains exactly r intervals in output. Now, we claim that for all $j = 1, \ldots, r-1$, we have $I_j^* \prec I_{j+1}^*$, *ie.* $re(I_j^*) < le(I_{j+1}^*)$. Indeed, assume that $I_j^* \equiv I_{u,j}$ and $I_{j+1}^* \equiv I_{v,j+1}$ with $u, v \in \{1, \ldots, t\}$. If $u = v$, the claim is proved. Otherwise, suppose that I_j^* and I_{j+1}^* are intersecting. We have $le(I_{v,j+1}) \leq re(I_{u,j})$ and also $re(I_{v,j}) < re(I_{u,j})$. Now, $I_{v,j+1} \in F$ at step $j+1$ implies necessarily $I_{v,j} \in F$ at step j. Then, $I_{v,j}$ has a smaller right endpoint than $I_{u,j} \equiv I_j^*$ in F at step j, which is a contradiction. $\square$

Now the proposition is proved as follows.

Proof of Proposition 1. Let $\mathcal{S} = \{S_1, \ldots, S_q\}$ be a partition of G into independent sets such that for all $i = 1, \ldots, q$, we have $|S_i| \geq k$. Define $|S_i| = \alpha_i k + \beta_i$ to be the size of an independent set S_i with α_i a non-zero integer and $\beta_i \in \{0, \ldots, k-1\}$. First, from each independent set S_i are extracted $\alpha_i - 1$ independent sets of size k, plus one if $\beta_i = 0$. After this preprocessing, at most $2k-1$ intervals remain in each independent set. Then, Lemma 1 is applied with $t = k$ to extract independent sets of size k while at least k independent sets of size strictly larger than k exist in the partition $\mathcal{S}$. When it remains less than k such independent sets in $\mathcal{S}$, a last application of Lemma 1 allows to conclude. The execution of Algorithm ExtractStable requiring $k\ O(t)$ time, the method described here runs in $k\ O(n/k) = O(n)$ time (given the intervals ordered according to $\prec$ in each input independent set). $\square$

Now, by a simple extension of Lemma 1, we establish that Proposition 1 holds for circular-arc graphs too. This extension relies on a simple fact: by removing the arcs which contain any point p on the circle, a circular-arc graph becomes an interval graph. Thus, the previous algorithm can be modified as follows to be still correct for circular-arc graphs. Select any point p on the circle. Moving clockwise once around the circle, define $I_{i,j}$ to be the j^{th} arc (become interval) following the arc containing p. Then, use Algorithm ExtractStable to obtain a solution (the right endpoint becomes the clockwise endpoint). The proof of the correctness remains the same since the arcs containing the point p induces a clique. Indeed, we always have $I_j^* \prec I_{j+1}^*$ for all $j = 1, \ldots, r-1$ and in addition, $I_r^* \cap I_1^* = \emptyset$ (the contrary would imply that I_r^* contains the point p). Finally, Lemma 1 is always applyable for circular-arc graphs and the following proposition is established.

Proposition 2 *Let G be a circular-arc graph with n vertices and an integer k. If G admits a coloring such that each color is used at least k times, then $\chi(G,k) = \lceil n/k \rceil$. Moreover, a partition of G into independent sets of size at most k can be computed in $O(n)$ time and $O(n)$ space given an ordered circular-arc representation and the coloring in input.*

Remark. Ordered interval (or circular-arc) representations can be computed in linear time by using recognition algorithms for interval (or circular-arc) graphs (see [6, 15] and [21]).

Corollary 1 *Let G be an interval graph (or a circular-arc graph) with n vertices. If $\chi(G,k) = \lceil n/k \rceil$ then $\chi(G,k') = \lceil n/k' \rceil$ for all integer $k' < k$.*

3 Extensions for triangulated graphs

In this section, we extend Proposition 1 to the class of triangulated graphs for values $k = 2, 3$. We shall denote with $d(v)$ the degree of a vertex v.

Proposition 3 *Let G be a triangulated graph with n vertices and an integer $k \leq 3$. If there exists a coloring of G such each color is used at least k times, then $\chi(G,k) = \lceil n/k \rceil$.*

The following lemma establishes immediately the proposition for $k = 2$.

Lemma 2 *Let A and B two independent sets of size 2 in G. Then there exists an independent set of size 2 having exactly a vertex in A and a vertex in B.*

Proof. Let $A = \{a_1, a_2\}$ and $B = \{b_1, b_2\}$. Suppose that there is no independent set such that it is described in the lemma. This implies that a_1 is connected to b_1 and b_2 and a_2 is connected to b_1 and b_2 too. In this case, $\{a_1, b_1, a_2, b_2, a_1\}$ is a cycle of length four without a chord, which is a contradiction. $\square$

For $k = 3$, the proof relies on the following lemmas.

Lemma 3 *Let $H = (X, Y, E)$ be a triangulated bipartite graph with $|X| + |Y| = n$ and $|E| = m$. Then we have the (in)equalities $\sum_{x\in X} d(x) = \sum_{y\in Y} d(y) = m \leq n - 1$.*

Proof. The equality $\sum_{x\in X} d(x) = \sum_{y\in Y} d(y) = m$ holds because H is bipartite. Moreover, a triangulated bipartite graph is isomorphic to a tree (a bipartite graph cannot contain a cycle of length three). Hence, we have $m \leq n - 1$. $\square$

Lemma 4 *Let $H = (A, B, E)$ be a triangulated bipartite graph such that $|A| = |B| = 4$. There exists in H an independent set of size 4 having exactly two vertices in A and two vertices in B.*

Proof. Suppose that such an independent set does not exist and set $A = \{a_1, a_2, a_3, a_4\}$ and $B = \{b_1, b_2, b_3, b_4\}$. We claim that for all distinct pairs $a_i, a_j \in A$, we have $d(a_i) + d(a_j) \geq 3$. Indeed, the contrary implies that there are at least two vertices in B which are not connected to a_i and a_j and also implies the existence of an independent set of size 4 with two vertices in A and two vertices in B. According to "pigeon hole" principle, this claim imposes to A to contains three vertices of degree at least 2. On the other hand, Lemma 3 imposes to the sum of degrees in A to be smaller than 7. Thus, without loss of generality, we can set $d(a_1) \leq 1$, $d(a_2) \geq 2$, $d(a_3) \geq 2$ and $d(a_4) \geq 2$. Now, by applying Lemma 3 to the bipartite subgraph induced by $A \setminus \{a_1\}$ and B, we obtain the stronger condition $d(a_2) + d(a_3) + d(a_4) \leq 6$ which implies (a) $d(a_2) = d(a_3) = d(a_4) = 2$. By using a similar proof with the b_i's, we establish that (b) $d(b_1) = d(b_2) = d(b_3) = 2$. Therefore, by supposing that $d(a_1) = d(b_4) = 1$ (in the worst case), the conditions (a) and (b) implies that H is isomorphic to a chain (see Fig. 3).

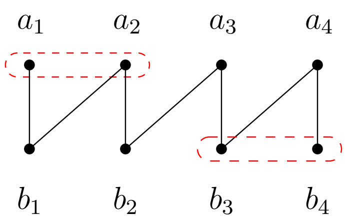

Fig. 3. The chain.

In this case, $\{a_1, a_2, b_3, b_4\}$ induces an independent set of size 4 with two vertices in A and two vertices in B, which is in contradiction with our first hypothesis. $\square$

Lemma 5 *Let $H = (A \cup B, C, E)$ be a triangulated bipartite graph such that $|A| = |B| = 2$ and $|C| = 4$. There exists in H an independent set of size 3 having exactly one vertex in A, one vertex in B and one vertex in C.*

Proof. Suppose that such an independent set does not exist and let $A = \{a_1, a_2\}$, $B = \{b_1, b_2\}$ and $C = \{c_1, c_2, c_3, c_4\}$. First, we claim that for all pairs a_i, b_j, we have $d(a_i) + d(b_j) = 4$ (the contrary implies the existence of a desired independent set). By summing on all the pairs a_i, b_j, we obtain

$d(a_1) + d(a_2) + d(b_1) + d(b_2) = 8$. This is in contradiction with the condition $d(a_1) + d(a_2) + d(b_1) + d(b_2) \leq 7$ imposed by Lemma 3. Consequently, our hypothesis is false and the desired independent set exists. $\square$

By using the proof of Proposition 1 with the following lemma, we establish definitely Proposition 3 for $k = 3$.

Lemma 6 *Let G be a triangulated graph.*

(1) If A, B, C are three independent sets of G of size 4, then there exists an independent set of size 3 having exactly one vertex in A, one vertex in B and one vertex in C,

(2) If A, B are two independent sets of G respectively of size 5 and 4, then there exists an independent set of size 3 having exactly two vertex in A and one vertex in B.

(3) If A, B are two independent sets of G respectively of size 4 and 4, then there exists an independent set of size 2 having exactly one vertex in A and one vertex in B.

Proof. The first assertion is proved by applying successively Lemmas 4 and 5 with independent sets A, B, C. The second and third assertions are directly deduced from Lemma 4 (or simply by Lemma 2 for the third one). $\square$

The proof technique used to establish Proposition 3 is essentially based on the fact that a triangulated bipartite graph is isomorphic to a tree and also is not a dense graph. That is why we conjecture that Proposition 3 holds for any integer k.

4 Counter-examples for bounded tolerance and co-comparability graphs

To conclude, we show that Proposition 1 cannot be extended to bounded tolerance graphs and also to co-comparability graphs. In effect, the graph $K_{k+1,k+1}$ is a counter-example for any integer k. $K_{k+1,k+1}$ is the complete bipartite graph on $2k+2$ vertices partitioned into two $k+1$-independent sets.

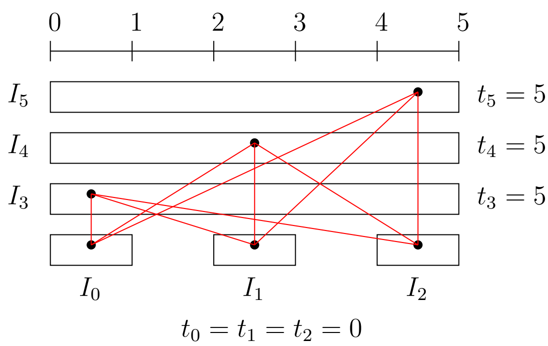

Fig. 4. The bounded tolerance graph $K_{3,3}$ for $k = 2$.

Such a graph has the following bounded tolerance representation (see Fig. 4): define $k+1$ intervals $I_0 = [0,1], I_1 = [2,3], \ldots, I_k = [2k, 2k+1]$ with tolerances $t_0 = t_1 = \cdots = t_k = 0$ and $k+1$ intervals $I_{k+1} = I_{k+2} = \cdots = I_{2k+1} = [0, 2k+1]$ with tolerances $t_{k+1} = t_{k+2} = \cdots = t_{2k+1} = 2k+1$. It is easy to show that $\chi(K_{k+1,k+1}, k) = 4$ since two vertices of different independent sets cannot be matched. Finally, this bound is strictly larger than the lower bound $\lceil n/k \rceil = \lceil (2k+2)/k \rceil = 3$.

References

[1] B.S. Baker and E.G. Coffman (1996). Mutual exclusion scheduling. *Theoretical Computer Science* 162, 225–243.

[2] Bamboo–Planification by Prologia–Groupe Air Liquide. http://prologianet.univ-mrs.fr/bamboo/bamboo_planification.html

[3] C. Berge (1985). *Graphs.* Elsevier Science Publishers B.V., Amsterdam, 2nd edition.

[4] J. Blazewick, K.H. Ecker, E. Pesch, G. Schmidt and J. Weglarz (1996). *Scheduling computer and manufacturing processes.* Springer, Heidelberg.

[5] H.L. Bodlaender and K. Jansen (1995). Restrictions of graph partition problems. Part I. *Theoretical Computer Science* 148, 93–109.

[6] K.S. Booth and G.S. Lueker (1976). Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *Journal of Computer and System Sciences* 13, 335–379.

[7] F. Gardi (2002). The mutual exclusion scheduling for proper interval graphs. Technical Report 02-2002, Laboratoire d'Informatique Fondamentale de Marseille.

[8] M.R. Garey, D.S. Johnson, G.L. Miller and C.H. Papadimitriou (1980). The complexity of coloring circular arcs and chords. *SIAM Journal on Algebraic and Discrete Methods* 1(2), 216–227.

[9] F. Gavril (1972). Algorithms for minimum coloring, maximum clique, minimum covering by cliques, and maximum independent set of a chordal graph. *SIAM Journal on Computing* 1, 180–187.

[10] P.C. Gilmore and A.J. Hoffman (1964). A characterization of comparability graphs and of interval graphs. *Canadian Journal of Mathematics* 16, 539–548.

[11] M.C. Golumbic (1980). *Algorithmic Graph Theory and Perfect Graphs.* Computer Science and Applied Mathematics. Academic Press, New-York.

[12] M.C. Golumbic and A.N. Trenk (2003). *Tolerance graphs.* monograph, in preparation.

[13] M.C. Golumbic and C.L. Monma (1982). A generalization of interval graphs with tolerances. In *Proc. 13th Southeastern Conference on Combinatorics, Graph Theory and Computing*, pages 321–331, *Congressus Numerantium* 35.

[14] U.I. Gupta, D.T. Lee and J.Y.-T. Leung (1982). Efficient algorithms for interval and circular-arc graphs. *Networks* 12, 459–467.

[15] M. Habib, R. McConnel, C. Paul and L. Viennot (2000). Lex-BSF and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing. *Theoretical Computer Science* 234, 59–84.

[16] S. Irani and V. Leung (1996). Scheduling with conflicts, and applications to traffic signal control. In *Proc. SIAM Symposium on Discrete Algorithms*, pages 85–94, SODA 96.

[17] K. Jansen (1998). The mutual exclusion scheduling problem for permutation and comparability graphs. In *Symposium on Theoretical Aspects of Computer Science*, pages 287–297, *LNCS* 1373, STACS 98.

[18] R.M. Karp (1972). Reduciblity among combinatorial problems. In R.E. Miller and J.W. Thatcher (Eds.), *Complexity of Computer Computations*, pages 85–103, Plenum, New-York.

[19] J. Krarup and D. de Werra (1982). Chromatic optimization: limitations, objectives, uses, references. *European Journal of Operations Research* 11, 1–19.

[20] Z. Lonc (1991). On complexity of some chain and antichain partition problem. In *Graph Theoretical Concepts in Computer Science*, pages 97–104. *LNCS* 570, WG 91.

[21] R.M. McConnel (2001). Linear time recognition of circular-arc graphs. In *Proc. IEEE Symposium on Foundations of Computer Science*, pages 386–394, FOCS 01.

[22] S. Olariu (1991). An optimal greedy heuristic to color interval graphs. *Information Processing Letters* 37, 21–25.

[23] F.S. Roberts (1978). *Graph Theory and its Applications to Problems of Society*. SIAM, Philadelphia, 1978.