Characterizing some Classes of Perfect Graphs through an Edge Classification

Stavros D. Nikolopoulos

08-97

Technical Report No. 08-97/1997

Department of Computer Science
University of Ioannina
45 110 Ioannina, Greece

Characterizing some Classes of Perfect Graphs through an Edge Classification

Stavros D. Nikolopoulos

Department of Computer Science, University of Ioannina, P.O. Box 1186, GR-45110 Ioannina, Greece.
e-mail: stavros@cs.uoi.gr

Abstract — We propose a novel classification of the edges of an undirected graph and show that this classification can be used as a constructive tool in proving structural and recognition properties for several classes of perfect graphs. Specifically, we classify the edges of a graph as either *free*, *semi-free* or *actual* and we define the class of *free* graphs as the class containing all the undirected graphs with no actual edges. We prove that the free graphs satisfy several important properties and are characterized by specific forbidden induced subgraphs. Based on these results, we show the relationship between free graphs and the classes of perfect graphs known as domination perfect, chordal (or triangulated), cographs, comparability, interval, permutation, ptolemaic, distance-hereditary, block, split and threshold. Consequently, we show that free graphs can be efficiently recognized in parallel by examining the closed neighbourhoods of the end-vertices of their edges, which, in turn, implies constant-time parallel recognition algorithms for all the above mentioned classes of perfect graphs in the case where their input graphs contain no actual edges.

1. Introduction

An undirected graph G = (V, E) is said to be *perfect* if it satisfies the following two properties: the χ-Perfect property: $\chi(G_A) = \omega(G_A)$ (for all $A \subseteq V$), and the α-Perfect property: $\alpha(G_A) = \varkappa(G_A)$ (for all $A \subseteq V$), where $\chi(G_A)$, $\omega(G_A)$, $\alpha(G_A)$ and $\varkappa(G_A)$ are the chromatic, clique, stability and clique-cover number of G_A, respectively, and G_A is an induced subgraph of G.

Our objective is to study recognition properties for some important classes of perfect graphs known as *domination perfect, chordal* (or triangulated), cographs, comparability, interval, permutation, ptolemaic, distance-hereditary, block, split and threshold graphs. Many researchers have extensively studied these classes of perfect graphs and proposed algorithms for the recognition problem, as well as for many other problems such as colouring, minimal code-colouring, maximal matching, clique finding, constructing perfect elimination schemes, assigning transitive orientations, clustering, assigning transitive orientations, minimum weight domination, minimal path cover, isomorphism, etc. (see, e.g., [11, 29]).

In this paper, we introduce an edge classification and show that it can be used as a constructive tool in proving recognition properties for the most important classes of perfect graphs. Based on this classification, we define the class of *free* graphs as the class which contains all the undirected graphs having no actual edges. We show structural properties and characterizations of the members of this class, which imply that free graphs form a subclass of chordal, cographs, ptolemaic, distance-hereditary, comparability, interval and permutation graphs. Moreover, we show recognition properties for block graphs, split graphs and threshold graphs, still using the proposed edge classification [14, 25, 27].

Specifically, given an undirected graph, we partition the edges of the graph into three classes, called *free*, *semi-free* and *actual* edges, according to the relationship of the closed neighbourhoods of the endpoints (or end-vertices) of their edges. Consequently, we prove that any free graph, i.e., any graph with no actual edges possesses, among others, the following important properties: Chordality or property T; a graph satisfying T is said to be chordal or triangulated; Transitive orientation or property C; a graph satisfying C is said to be comparability; Transitive co-orientation or property C^c; a graph satisfying C^c is said to be co-comparability, i.e., its complement is a comparability graph; Clique-kernel intersection property or CK property. Moreover, based on the definition of the actual edges of a graph, we show that the free graphs are exactly the graphs not having a P_4 or a C_4 as an induced subgraph.

It is well-known that several classes of perfect graphs have already been characterized in terms of these properties, as well as in terms of forbidden induced subgraphs. For example, interval graphs satisfy properties T and C^c [12], permutation graphs satisfy properties C and C^c [28], cographs satisfy the CK property [4], cographs have no induced subgraphs isomorphic to P_4 [4], threshold graphs have no induced subgraph isomorphic to $2K_2$, P_4, or C_4 [5], etc. Based on these properties and characterizations, we show that free graphs belong to the classes of domination perfect, chordal, cographs, comparability, interval, permutation, ptolemaic and distance-hereditary graphs. Moreover, we identify the precise structure possessed by certain subsets of vertices and/or edges of a graph in the case where it is a block, split or threshold graph. Consequently, we formulate a constant-time parallel algorithm for deciding whether or not an undirected graph contains actual edges, which operates by examining specific relations of the closed neighbourhoods of the endpoints of each edge of the graph. This result, in turn, implies than all the above mentioned perfect graphs can be recognized in constant-time in the case where they contain no actual edges. The parallel algorithms proposed in this paper run on a Concurrent-Read, Concurrent-Write (CRCW) PRAM model of computation and use $O(m\,n)$ processors.

Throughout the paper we assume that all graphs are finite and that unless stated otherwise the term subgraph always refers to the notion of induced subgraph. Moreover, m denotes the number of edges and n denotes the number of vertices in a graph.

2. Free Graphs

Following the notation and terminology in [13, p.167], the *neighbourhood* of a vertex u is the set $N(u)$ consisting of all the vertices v which are adjacent with u. The *closed neighbourhood* is $N[u] = \{u\} \cup N(u)$.

Given a graph G = (V, E), we define three classes of edges in G, denoted by AE, FE and SE, according to relationship of the neighbourhood and closed neighbourhood of the endpoints of its edges [14, 25, 27]. Let $x = (u, v)$ be an edge of G. Then,

$$
\begin{array}{lll}
(u, v) \in \mathrm{FE} & \text{if} & N[u] = N[v] \\
(u, v) \in \mathrm{SE} & \text{if} & N[u] \subset N[v] \\
(u, v) \in \mathrm{AE} & \text{if} & N[u] \neq N[v]
\end{array}
$$

In words, edge (u, v) is a member of FE if its vertices u and v have the same closed neighbourhoods; it is a member of SE if the closed neighbourhood of vertex u (resp. v) is contained in the closed neighbourhood of vertex v (resp. u); it is a member of AE if the closed neighbourhoods of vertices u and v, i.e., $N[u]$ and $N[v]$, are not comparable with respect to inclusion. Specifically, an edge $(u, v) \in$ AE if both sets $N[u]$ and $N[v]$ have at least a common vertex, and the vertex set $N[u]$ (resp. $N[v]$) has at least one vertex which does not belong to $N[v]$ (resp. $N[u]$). An edge is said to be a *free*, *semi-free* and *actual* edge if it is a member of class FE, SE and AE, respectively. Obviously, E = FE + SE + AE. We illustrate with three graphs G, H and I shown in Fig. 1. The edges in classes FE, SE and AE are denoted by *f*, *s* and *a*, respectively.

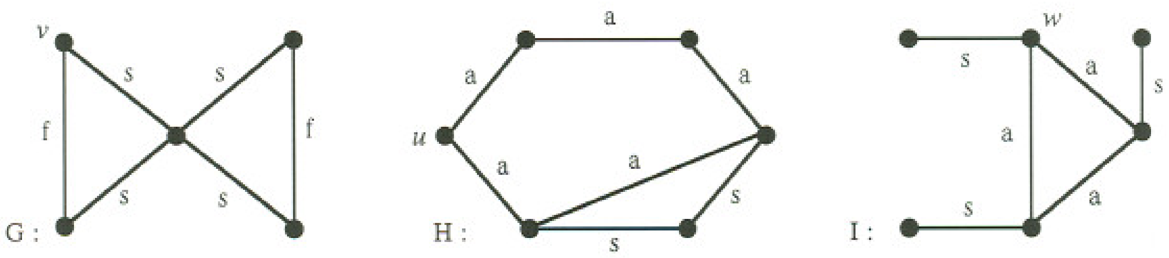

Figure 1. Three undirected graphs. Free, Semi-free and Actual edges are denoted by f, s and a, respectively.

In addition to the above, we define thre classes of vertices of a graph G = (V, E), denoted by FV, SV and AV, as follows: Vertex u belongs to FV (resp. AV) if it is an endpoint of a *free edge* (resp. *actual edge*), and u belongs to SV if there exists a vertex v such that $(u, v) \in$ SE and $N[u] \subset N[v]$. Similarly, we shall call *free, semi-free* and *actual* vertices the elements of the sets FV, SV and AV, respectively.

Having classified the edges of a graph as either free, semi-free and actual, let us now define the class of *free graphs* as follows:

Definition 1. A undirected graph G = (V, E) is called *free* if every edge of G is either free or semi-free edge.

The following results provide algorithmic properties for the class of free graphs, i.e., the undirected graphs with no actual edges.

Lemma 1. Let G = (V, E) be an undirected graph with no actual edges, i.e., AE = $\varnothing$. Then, there exists a partition of the vertex set V into nonempty, disjoint vertex sets $V_1, V_2, ..., V_m$, $m \geq 1$, i.e., $V = V_1 \cup V_2 \cup ... \cup V_m$, satisfying the following properties:

(i) If $x, y \in V_i$ then $(x, y) \in$ FE, $1 \leq i \leq m$.

(ii) If $x \in V_i$, $y \in V_j$ and $(x, y) \in$ E then $(x, y) \in$ SE, $i \neq j$ and $1 \leq i, j \leq m$.

Proof. Since AE = AV = $\varnothing$, there follows that E = FE $\cup$ SF and V = FV $\cup$ SV. If FE $\neq \varnothing$ and SE = $\varnothing$ there is nothing to prove since G is a complete graph K_n, i.e., V = FV = V_1. If FE = $\varnothing$ and SE $\neq \varnothing$ there is also nothing to prove since G has the following property: there exists a vertex $x \in$ V such that $N(x)$ = V - $\{x\}$ and every connected components of G(V-$\{x\}$) is a K_1, P_3 or induces a star graph on $m < n$ vertices (obviously, if every connected components is K_1 then G a star graph on n vertices $K_{1,n-1}$). In any case $V = V_1 \cup V_2 \cup ... \cup V_n$, where $| V_i | = 1$, $1 \leq i \leq n$. We consider now the case where both sets FE and SE are not empty (see graph G in Fig. 1). By definition, the set of free edges FE has the following property: if $(x, y) \in$ FE and $(y, z) \in$ FE then $(x, z) \in$ FE. It follows that the edge set FE has the form

$$FE = FE_1 \cup FE_2 \cup ... \cup FE_r$$

where the set FE_i, $1 \leq i \leq r$, is such that: if it contains a free edge (x, y) then it contains every other free edge having an endpoint on x or y. The corresponding vertex set FV of the edge set FE has the following form

$$FV = V_1 \cup V_2 \cup ... \cup V_r$$

where V_i contains all the vertices of G which form a free edge in FE_i, $1 \leq i \leq r$. From the above we conclude that, set V_i is a clique, $1 \leq i \leq r$, and if an edge $(x, y) \in$ E has an endpoint in V_i and the other endpoint in V_j, $i \neq j$, then $(x, y) \in$ SE. Therefore, the lemma is proved for the graph G(FV). Let us now focus on the properties of the elements of set SE. This set contains semi-free edges with an endpoint in V_i and the other in either V_j or V-FV, where $i \neq j$ and $1 \leq i \leq r$. If FV = V the lemma holds. In the case where FV $\subset$ V, every vertex in V-FV joined by a semi-free edge with a vertex in at least one set V_i, $1 \leq i \leq r$. This implies that we can partition the vertex set V-FV into one-vertex disjoint sets $V_{r+1}, V_{r+2}, ..., V_m$, $m \leq n$, i.e., $| V_i | = 1$ for $i = r+1, ..., m$. Hence, the lemma follows. □

Lemma 2. Let G = (V, E) be an undirected graph with no actual edges, i.e., AE = $\varnothing$, and let $V = V_1 \cup V_2 \cup ... \cup V_m$, $3 \leq m \leq n$, be a partition of V satisfying the properties of Lemma 1. If $N(V_i) \cap V_j \neq \varnothing$, then $V_i \cup V_j$ induces a clique, $i \neq j$.

Proof. Let V_i, V_j be two vertex sets satisfing the condition $N(V_i) \cap V_j \neq \varnothing$, $i \neq j$, and let $p \in V_i$ and $q \in V_j$ be such that $(p, q) \in$ E. Since $(p, q) \in$ SE (see Lemma 1) we may suppose that $N[p] \subset N[q]$. Also by Lemma 1 all edges in sets V_i and V_j belong to FE. This implies:

$$\forall\, y \in V_j : V_i \subseteq N[p] \subset N[q] = N[y]$$

saying that the vertex set $V_i \cup V_j$ induces a clique. □

Lemma 3. Let G = (V, E) be an undirected graph with no actual edges, i.e., AE = $\emptyset$, and let $V = V_1 \cup V_2 \cup ... \cup V_m$, $3 \le m \le n$, be a partition of V satisfying the properties of Lemma 1. Then, there exists a clique V_k, $1 \le k \le m$, satisfying the following property:

$$N(V_k) = V_1 \cup V_2 \cup ... \cup V_{k-1} \cup V_{k+1} \cup ... \cup V_m$$

where $m \ge 1$.

Proof. As a preliminary to proof of this Lemma, we can define a transitive ordering < on the clique set $\{V_1, V_2, ..., V_m\}$ on the bases of:

$$V_i < V_j \iff \exists\, p \in V_i,\ q \in V_j :\ (p, q) \in E,\ N[p] \subset N[q]$$

This is in fact equivalent to:

$$\forall\, x \in V_i,\ y \in V_j :\ N[x] \subset N[y]$$

which holds due to Lemma 2, and because all edges in V_i and V_j belong to FE. The ordering can be represented by an oriented connected graph in which the (hyper)vertices correspond with cliques in the original graph.
Now it is easily verified that the ordering allows a unique largest element, which conforms with the clique V_k (if not, there should be cliques V_r, V_s, V_t, where $r \ne s \ne t \ne r$, such that $V_r < V_s$ and $V_r < V_t$ which is impossible). □

Corollary 1. Let G = (V, E) be a graph with no actual edges and let $V = V_1 \cup V_2 \cup ... \cup V_m$, $3 \le m \le n$, be a partition of V satisfying the properties of Lemma 1. Then, $V_k \cup V_i$ induces a clique, for $i = 1, 2, ...k\text{-}1, k\text{+}1, ..., m$.

Lemma 4. Every subgraph of a free graph is a free graph.

Proof. The lemma is obviously true since every subgraph of a free graph contains no actual edges. □

A typical structure of a free graph is shown in Fig. 2. The results of this section can be summarized as follows:

Theorem I (see also [27]). The vertex set V of a free graph G = (V, E) can be partitioned into $m \ge 2$ nonempty, disjoint vertex sets $V_1, V_2, ..., V_k, ..., V_m$, i.e.,

$$V = V_1 \cup V_2 \cup ... \cup V_k \cup ... \cup V_m,$$

satisfying the following properties:

(P1) There exists a vertex set V_k such that $N[V_k] = V$, $1 \le k \le m$.

(P2) Every vertex set V_i induces a complete graph $G(V_i)$, i.e., V_i is a clique, $1 \le i \le m$.

(P3) Every vertex set $V_i \cup V_j$ induces either a complete graph $G(V_i \cup V_j)$, or a disconnected graph having two complete subgraphs $G(V_i)$ and $G(V_j)$, $1 \le i, j \le m$.

(P4) Edges with both endpoints in V_i are free edges, $1 \le i \le m$.

(P5) Edges with one endpoint in V_i and the other endpoint in V_j are semi-free edges, $1 \le i, j \le m$ and $i \ne j$.

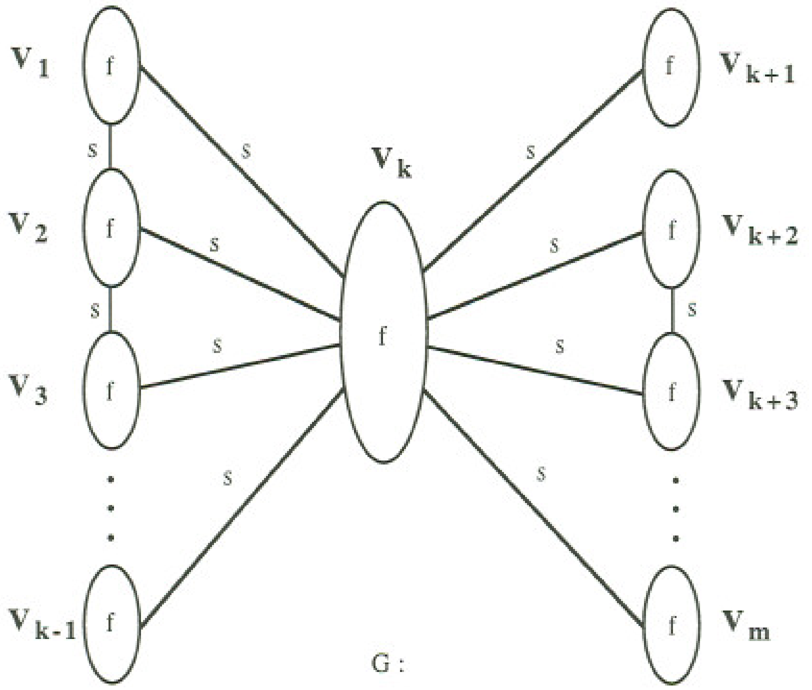

Figure 2. The typical structure of a free graph. A line between cells V_i and V_j indicates that each vertex in V_i is adjacent to each vertex of V_j. All edges in V_i are free edges; All edges between cells are semi-free edges.

The graph G in Fig. 1 is a free graph, while the graphs H and I in the same figure are not free graphs. Next, we prove that the free graphs satisfy important properties which are later used as a base for showing the relationship between the class of free graphs and many other classes of perfect graphs.

A graph is a *diagonal graph* or *D*-graph if for every path in G with edges (v_1, v_2), (v_2, v_3), (v_3, v_4), the graph also contains the edges (v_1, v_3) or (v_2, v_4). It is important to point out that Wolk [34] showed that the *D*-graphs are precisely the comparability graphs of rooted trees. This result was later quoted incorrectly as "A graph without induced subgraph isomorphic to P_4, i.e., a cograph, is the comparability graph of rooted trees". The graph C_4 is a counter-example to this statement. From the definition, it is easy to see that *D*-graphs contain no actual edges. Therefore, we are in a position to state our first result.

Theorem 1. Diagonal graphs (or *D*-graphs) are precisely the undirected graphs with no actual edges, i.e., the free graphs.

Based on the definition of the actual edges of a graph, we can easily show that the free graphs are exactly the graphs not having a P_4 or a C_4 as an induced subgraph. Thus, the following theorem holds.

Theorem 2. A free graph G contains no induced subgraph isomorphic to P_4 or C_4.

A *sun* of order p, or *p-sun* ($p \geq 3$) is a chordal graph on vertex set $\{x_1, x_2, ..., x_p, y_1, y_2, ..., y_p\}$, where $\{y_1, y_2, ..., y_p\}$ is an independent set, $(x_1, x_2, ..., x_p)$ is a cycle, and each vertex y_i has exactly two neighbours, x_{i-1} and x_i. By definition, every *p-sun* ($p \geq 3$) contains an actual edge. So, we obtain the following results.

Theorem 3. Let G be free graph. Then, G contains no induced subgraph isomorphic to p-sun ($p \geq 3$).

For a graph G the k-th power G^k of G is the graph with the same vertex set as G where two vertices are adjacent if and only if their distance is at most k in G. The *clique graph* K(G) of G is the graph whose vertices are the maximal cliques $K^1, K^2, \ldots, K^p$ of G, in which (K^i, K^j) is an edge if and only if $K^i \cap K^j \neq \emptyset$, where $i \neq j$. The following theorem clarify the relationship between G^2 and K(G) of a free graph G.

Theorem 4. Let G be free graph. Then, both G^2 and K(G) are complete graphs.

Let $\gamma(G)$ and $\iota(G)$ be the domination number and independent domination number of a graph G, respectively. A graph G is called *domination perfect* graph if $\gamma(H) = \iota(H)$, for every induced subgraph H of G. The domination number $\gamma(G)$ is the minimum cardinality taken over all dominating sets of G, and the independent domination number $\iota(G)$ is the minimum cardinality taken over all maximal independent sets of vertices of G. Based on the properties (P1) and (P2) of Theorem I, we can prove $\gamma(H) = \iota(H) = 1$, for every induced subgraph H of a free graph G. Thus, we obtain the following theorem.

Theorem 5. Let G be free graph. Then, G is a domination perfect graph.

3. Relationship between Free and Perfect Graphs

Based on the previous results, we show here that the classes of perfect graphs known as chordal, cographs, ptolemaic, distance-hereditary, comparability, interval and permutation, properly contain the class of free graphs.

3.1. Chordal Graphs and Cographs

A graph G = (V, E) is called *chordal* (or triangulated) if every cycle of length, at least, four has a chord, i.e., an edge joining two nonconsecutive vertices on a cycle [7, 19, 20, 24]. Chordal graphs arise in the study of Gaussian elimination on sparse symmetric matrices, in the study of acyclic relational schemes, and are related to and useful for many location problems [15, 26, 30, 32]. *Cographs* (or complement reducible graphs) are defined as the class of graphs formed from a single vertex under the closure of the operations of union and complement. Cographs is a well-know class of perfect graphs arising in many disparate areas of mathematics and computer science such as scheduling, colouring, computational semantics, and other practical applications. Cographs themselves were introduced in the early 1970s by Lerchs [21] who studied the structural and algorithmic properties of these graphs and showed the following two very nice algorithmic properties (see also [4, 6, 22, 31]): (i) cographs are exactly the P_4 restricted graphs, and (ii) a cograph has a unique tree representation called cotree. Next, we present the fundamental theorem on cographs.

Theorem II (Corneil, Perl and Stewart [1985]). Let G = (V, E) be a undirected graph. Then, the following statements are equivalent:

(i) G is a cograph;

(ii) G does not contain P_4 as a subgraph;

(iii) any subgraph of G has the CK-property;

An immediate consequence of the results provided by Lemma 2.1 and statements (i) and (ii) of Theorem II is that free graphs are exactly the chordal cographs. Thus, the following theorem holds.

Theorem 6. Let G = (V, E) be free graph. Then, G is both chordal graph and cograph.

Let us comment on statement (iii). A *kernel* of a graph is a maximal independent set and a *clique* is a maximal complete set (in fact such a clique is a *maximal clique*). Obviously, the vertex set $S \subseteq V$ is a kernel in G if and only if S is a clique in G^c. A graph is said to have the *clique-kernel intersection property* (or CK-property) if and only if every clique of G has one vertex in common with every kernel of G. Obviously, every subgraph of a free graph is a free graph and, therefore, every subgraph of a free graph has the CK-property.

A graph G is called *strongly chordal* if G is chordal and G contains no sun, G is called *balanced chordal* if G is chordal and G contains no sun of odd order, and G is called *compact* if G contains no sun of order 3. We have showed that a free graph is a chordal graph (Theorem 6) and it contains no induced subgraph isomorphic to p-sun, $p \geq 3$ (Theorem 3). These prove the following result.

Theorem 7. Let G be free graph. Then, G is a strongly chordal graph, a balanced chordal graph and a compact graph.

We know the following three statements are equivalent for a chordal graph G: (i) G^2 is chordal; (ii) K(G) is chordal; (iii) every sun of G of order greater than 3 is suspended [33]. If G is a free graph, then G^2 and K(G) are complete graphs and, therefore, chordal graphs. Thus, we have the following result.

Theorem 8. Let G be free graph. Then, both G^2 and K(G) are chordal graphs and every sun of G of order greater than 3 is suspended.

3.2. Comparability and Permutation Graphs

The class of perfect graphs known as *comparability* (or transitive orientable) graphs, plays an important roll in graph theory, especially due to its close relation with the classes of interval and permutation graphs [10, 12]. An undirected graph G = (V, E) belongs to the class of comparability graphs if each edge of the graph can be assigned an one-way direction in such a way that the resulting oriented graph $G_f = (V, F)$ satisfies the following condition: $\langle x, y\rangle \in F$ and $\langle y, z\rangle \in F$ imply $\langle x, z\rangle \in F$, for every vertex $x, y, z \in V$.

We now consider a class of perfect graphs known as permutation graphs. A graph G = (V, E), with vertices numbered from 1 to n, i.e., V = {1, 2, ..., n}, is called a *permutation graph* if there exists a permutation $\pi = [\pi_1, \pi_2, ..., \pi_n]$ on $N = \{1, 2, ..., n\}$ such that,

$$(i, j) \in E \Leftrightarrow (i - j)\, (\pi^{-1}(i) - \pi^{-1}(j)) < 0$$

for all $i, j \in N$, where π_i^{-1}, denoted here as $\pi^{-1}(i)$, is the index of the element $i \equiv \pi_i$ in π [11, 28]. Pnueli, Lempel and Even established in 1971 (see [28]) where an undirected graph belongs to the class of permutation graphs.

Theorem III (Pnueli, Lempel and Even [1971]). A graph G is a permutation graph if and only if G is both comparability and cocomparability graph.

We have proved that a free graph is a cograph. Since cographs are a subclass of permutation graphs, cographs are comparability and cocomparability graphs. Moreover, it is well-known that a comparability graph is *superperfect* [11]. Therefore, the following theorem and its corollary hold.

Theorem 9. Let G = (V, E) be a free graph. Then, G is both comparability and cocomparability graph.

Corollary 2. A free graph is a permutation graph and superperfect graph.

3.3. Interval Graphs

An undirected graph G = (V, E) is called an *interval* graph if its vertices can be put into one-to-one correspondence with a set of intervals I of a linearly ordered set (like the real line) such that two vertices are connected by an edge of G if and only if their corresponding intervals have nonempty intersection [9, 11].

The following theorem establishes, through the properties T and C^c, where an undirected graph belongs to the class of interval graphs.

Theorem IV (Gilmore and Hoffman [1964]). An undirected graph G is a interval graph if and only if G is a chordal graph and its complement G^c is a comparability graph.

We have proved that a free graph is both chordal graph (Theorem 6) and cocomparability graph (Theorem 9). This result implies that free graphs are interval graphs. Moreover, it is well-known that every interval graph is a *circular-arc* graph; the converse, however, is false [11] (The intersection graphs obtained from collection of arcs on a circle are called circular-arc graphs.). Thus, the following theorem and its corollary hold.

Theorem 10. Let G = (V, E) be a free graph. Then, G is an interval graph.

Corollary 3. Free graphs form a subclass of circular-arc graphs.

3.4. Distance-hereditary and Ptolemaic Graphs

A graph G = (V, E) is called *distance-hereditary* if it is a connected graph in which every induced path is isometric. That is, the distance of any two vertices in an induced path equals their distance in the graph. Simply, a distance-hereditary graph G is a graph preserving distances in each connected induced subgraph [17, 18]. Distance-hereditary graphs were introduced by Howorka [17], who gave, among others, the following characterization of such graphs: a connected graph G is distance-hereditary if and only if every cycle in G of length at least 5 has a pair of chords cross each other. The distance-hereditary graphs form a subclass of the parity graphs [1, 4, 23] (a graph G is a parity graph if every odd cycle of G of length at least 5 has two crossing chords).

An important class of perfect graphs, known as ptolemaic graphs, forms a subclass of the distance-hereditary graphs. Actually, a graph G is a *ptolemaic* graph if and only if it is chordal and distance-hereditary graph. Moreover, it is well known that the class of *block* graphs forms a subclass of the ptolemaic graphs. Therefore, we have the following string of concepts:

block graphs $\rightarrow$ ptolemaic graphs $\rightarrow$ distance-hereditary graphs $\rightarrow$ parity graphs

We have proved that free graphs are exactly the chordal cographs. Moreover, cographs form a subclass of the class of distance-hereditary graphs (each connected induced subgraph preserves distances), and therefore, they form a subclass of the class of parity graphs. Thus, we can present the following theorem and its corollary.

Theorem 11. Let G be a free graph. Then, G is a ptolemaic graph.

Corollary 4. Free graphs form a subclass of distance-hereditary and parity graphs.

3.5. Block Graphs

A graph G is called *block* graph if it is connected and every block (i.e., maximal 2-connected subgraph) is complete [2]. Howorka [18] offered the following purely metric characterization: a connected graph is a block graph if and only if its distance function d satisfies the four-point condition, i.e., for any four vertices u, v, x, y, the larger two of the distance sum

$$d(u, v) + d(x, y), \quad d(u, x) + d(v, y), \quad d(u, y) + d(v, x)$$

are equal.

Unfortunately, all the free graphs do not satisfy the above four-point condition. For example, the free graph $K_2 + 2K_1$ give distance sums 2, 2 and 3. The next theorem provide us with another type of metric characterization, namely, via forbidden isometric subgraphs.

Theorem V (Bandelt and Mulder [1986]). Let G be a connected graph with distance function d. Then, the following statements are equivalent:

(i) G is a block graph;
(ii) d satisfies the four-point condition;
(iii) neither K_4 minus an edge nor C_n with $n \geq 4$ is an isometric subgraph of G;

We focus on statements (i) and (iii) of Theorem V. By definition, a free graph does not contain subgraphs isomorphic to C_n with $n \geq 4$, and therefore, it does not contain C_n ($n \geq 4$) as an isomorphic subgraph. It is easy to see that a graph is a block graph if and only if it is chordal and each edge appears only in one clique.

Theorem 12. Let $G = (V, E)$ be a free graph and let $|V_k| = 1$, where V_k is a clique satisfying the properties of Lemma 1. Then, G is a block graph if and only if there exists no semi-free edge (x, y) in G such that $x, y \notin V_k$.

Proof. ($\Rightarrow$) Let u be the vertex of set V_k. Suppose that there exists a semi-free edge (x, y) in G such that such that $x, y \notin V_k$. This implies that $x \in V_i$ and $y \in V_j$ where $i \neq j$. Since (x, y) is a semi-free edge, there exists a vertex $z \in V_p$, where $p \neq i$ and $p \neq j$, having the property $(z, x) \in E$ (or $(z, y) \in E$). Obviously, (x, u) appears in more than one clique; an absurd. ($\Leftarrow$) It is easy to see that $N(z) = V_i \cup \{u\}$ for every $z \in V_i$ and $i \neq k$, where $u \in V_k$. Since u is a cutpoint and G is a chordal graph, there follows that G is a block graph. □

Theorem 13. Let $G = (V, E)$ be a free graph and let $|V_k| > 1$, where V_k is a clique satisfying the properties of Lemma 1. Then, G is a block graph if and only if G is a complete graph.

Proof. ($\Rightarrow$) Let (u, v) be a free edge such that $u, v \in V_k$. Suppose that G is not a complete graph, and let x, y be two vertices such that $(x, y) \notin E$. Then, it is easy to see that G contains an induced subgraph $K_2 + 2K_1$ (a K_4 minus an edge), i.e., $G(\{u, v, x, y\})$. Thus, edge (u, v) appears in more than one clique, and therefore, G is not a block graph; an absurd. ($\Leftarrow$) Obviously, G is a block graph. □

5.6. Split Graphs

An undirected graph $G = (V, E)$ is defined to be *split* if there is a partition $V = K + S$ of its vertex set V into a complete set K and a stable set S.

It is well know that split graphs are characterized in terms of the properties T and T^c, i.e., split graphs $\equiv T + T^c$ (see Foldes and Hammer [8]). That is, a graph G is a split graph if and only if G and its complement G^c are chordal graphs.

Theorem VI (Foldes and Hammer [1977]). Let G be a undirected graph. The following conditions are equivalent:

(i) G is a split graph;
(ii) G and G^c are chordal graphs;
(iii) G contains no induced subgraph isomorphic to $2K_2$, C_4, or C_5;

Unfortunately, free graphs do not satisfy the property T^c since the complement of a split graph is not always a chordal graph. For example, the complement of the graph $2K_2$, which is the graph C_4, obviously is not a chordal graph. Therefore, in the context of this work, statements (i) and (ii) seems not to give us any useful information. On the other hand, statements (i) and (iii) provide us with a characterization of split graphs in terms of forbidden induced subgraphs. It is easy to see that a free graph contains no induced subgraph isomorphic to C_4 or C_5 (see Theorem 2). Thus, we obtain the following result on split graphs.

Theorem 14. Let G be a free graph. Then, G is a split graph if and only if G contains no induced subgraph isomorphic to $2K_2$.

5.2. Threshold Graphs

The class of *threshold graphs*, a well-known class of perfect graphs, is defined to contain those graphs where stable subsets of their vertex sets can be distinguished by using a single linear inequality. Equivalently, a graph $G=(V,E)$ is threshold if there exists a threshold assignment $[\alpha, t]$ consisting of a labelling α of the vertices by non-negative integers and an integer threshold t such that:

$$\text{S is a stable set iff } \alpha(v_1) + \alpha(v_2) + \ldots + \alpha(v_p) \leq t$$

where $v_i \in S$, $1 \leq i \leq p$ and $S \subseteq V$.

Threshold graphs were introduced in 1973 by Chvátal and Hammer [5]. They were rediscovered and studied by other researchers, including Henderson and Zalcstein [16] (see also [11, 12, 27]). Threshold graphs have an interesting application to computing which is the synchronization of parallel processes. Specifically, they provide a simple programming technique which can be applied to let the computer system to prevent conflicts automatically and control the traffic of programs running and waiting.

We have seen that most of the classes of perfect graphs we consider are characterized by forbidden (isometric in some cases) subgraphs. Chvátal and Hammer [5] have characterized the threshold graphs as the graphs which contain no induced subgraphs isomorphic to $2K_2$, P_4 or C_4.

Theorem VII (Chvátal and Hammer [1973]). Let G be a undirected graph. Then, the following statements are equivalent:

(i) G is a threshold graph;

(ii) G has no induced subgraph isomorphic to $2K_2$, P_4, or C_4;

We have proved that a free graph contains no induced subgraph isomorphic to P_4 or C_4 (see Theorem 2). By combining these results with the results of Theorem VII, we obtain the following theorem.

Theorem 15. Let G be a free graph. Then, G is a threshold graph if and only if G contains no induced subgraph isomorphic to $2K_2$.

4. Parallel Recognition Algorithms

We present here parallel algorithms for recognizing free graphs and block graphs with no actual edges. The model of parallel computation used is the well-known Concurrent-Read, Concurrent-Write PRAM model (CRCW PRAM) [3, 27].

4.1. Recognizing Free Graphs

We now formulate and analyze a parallel algorithm for computing the classes of free, semi-free and actual edges, and therefore, recognizing whether or not a undirected graph G = (V, E) is a free graph. The algorithm operates as follows:

Step 1. Compute the edge sets FE, SE and AE of a graph G as follows:
for every edge $(u, v) \in E$ do in parallel

if $N[u] = N[v]$ then $FE \leftarrow \{(u, v)\}$;
if $N[u] \subset N[v]$ or $N[u] \supset N[v]$ then $SE \leftarrow \{(u, v)\}$;
if $N(u) \cap N(v) = \varnothing$ then $AE \leftarrow \{(u, v)\}$; $AV \leftarrow \{u, v\}$;

Step 2. If $AV = \varnothing$, then G is a free graph;

The computational time-processor complexity of the above algorithm can be easily computed on a parallel computational model. Each operation of the algorithm is executed in $O(1)$ time with $O(n)$ processors, and therefore, the algorithm is executed in $O(1)$ time with $O(m\, n)$ processors using a CRCW PRAM. Thus, we obtain the following results.

Theorem 16. The free, semi-free and actual edges of an undirected graph G with n vertices and m edges can be computed in $O(1)$ time with $O(m\, n)$ processors on a CRCW PRAM model.

Corollary 5. Free graphs can be recognized in $O(1)$ time by using $O(m\, n)$ processors on a CRCW PRAM model.

Corollary 6. Domination perfect, chordal, strongly chordal, balanced chordal, compact, cographs, ptolemaic, distance-hereditary, parity, comparability, superperfect, interval, circular-arc and permutation graphs with no actual edges can be recognized in $O(1)$ time by using $O(n\, m)$ processors on a CRCW PRAM model.

4.2. Recognizing Block Graphs having no Actual Edges

The results provided by Theorems 12 and 13 implies a parallel algorithm for recognizing the subclass of block graphs whose members contain no actual edges. The recognition algorithm operates as follows:

Step 1. Compute the vertex set V_k of a free graph;
Step 2. If $|V_k| = 1$ then G is a block graph if there exists no semi-free edge (x, y) in G such that $x, y \notin V_k$.;
Step 3. If $|V_k| > 1$ then G is a block graph if G is a complete graph;

Having computed the computational complexity of recognizing free graphs, let us now compute the time and processor complexity of recognizing block graphs in case they contain no actual edges. Let G = (V, E) be a free graph and let A be the adjacency matrix of the graph G. A vertex $x \in V_k$ if and only if $A[x, y] = 1$ for every $y \neq x$. Therefore, the vertex set V_k can be computed in $O(1)$ time with $O(n^2)$ processors. Moreover, $|V_k| = 1$ if and only if there is only one vertex $x \in V_k$ having the property $A[x, y] = 1$ for every $y \neq x$. Thus, the operation of testing whether V_k contains one or more vertices can be executed in $O(1)$ with $O(n^2)$ processors. Similarly, the operation of testing whether a vertex set induces a complete graph or not can be executed in $O(1)$ with $O(n^2)$ processors. Based on Theorems 12 and 13, we can present the following result.

Theorem 17. Block graphs with no actual edges can be recognized in $O(1)$ time by using $O(n\, m)$ processors on a CRCW PRAM model.

4.3. Recognizing Split and Threshold Graphs having no Actual Edges

The characterizations provided by Theorems 14 and 15 offer information on how to design a constant-time parallel recognition algorithm for split and threshold graphs in the case where these graphs contain no actual edges. The recognition algorithm operates as follows:

Step 1. Compute the vertex set V_k of a free graph;
Step 2. Select semi-free edge (v, u) of $G(V-V_k)$;
Step 3. If there exists a semi-free edge (x, y) in $G(V-V_k)$ such that $\{N(x) \cup N(y)\} \cap \{v, u\} = \varnothing$ then G is not a split graph nor a threshold graph;

We observe that all the operations of the algorithm have been appeared in the previous algorithms and, thus, these operations have known time-processor complexity. We can therefore present the following results.

Theorem 18. Split and threshold graphs containing no actual edges can be recognized in $O(1)$ time by using $O(n\,m)$ processors on a CRCW PRAM model.

5. Conclusions

In this paper we classified the edges of a graph as either free, semi-free or actual, we defined the class of free graphs, i.e., the class of all the graphs with no actual edges and we proved that the members of this class possess several important properties among which the properties T, C and C^c, as well as the clique-kernel intersection property. Moreover, we showed that the free graphs are characterized by specific forbidden induced subgraphs. Based on the fact that many classes of perfect graphs are characterized in terms of these properties and forbidden induced subgraphs, we proved that free graphs belong to the class of domination perfect, chordal (or triangulated), cographs (or complement reducible), ptolemaic, distance-hereditary, comparability, interval and permutation graphs. The recognition of a free graph can be easily done in constant-time by using a powerful parallel model of computation. Furthermore, recognition properties for block, split and threshold graphs containing no actual edges have been also shown, leading to a constant-time parallel recognition algorithm.

We are currently studying other recognition properties and characterizations of free graphs in order to extend classes of perfect and/or non perfect graphs in which they might belong. We hope our study will also enable us to further extend classes of perfect graphs whose members can be recognized in parallel constant-time.

References

[1] G.S. Adhar and S. Peng, Parallel algorithms for cographs and parity graphs with applications, *J. of Algorithms* 11 (1990) 252-284.

[2] H-J Bandelt and H.M. Mulder, Distance-hereditary graphs, *J. of Comb. Theory: Series B*, 41 (1986) 182-208.

[3] P. Beame and J. Hastan, Optimal bounds for decision problems on the CRCW PRAM, *J. Assoc. Comput. Mach.* 36 (1989) 643-670.

[4] D.G. Corneil, Y. Perl and L.K. Stewart, A linear recognition algorithm for cographs, *SIAM J. Comput.* 14 (1985) 926-934.

[5] V. Chvátal and P.L. Hammer, Set-packing and threshold graphs, Res. Report CORR-73-21, University of Waterloo, 1973.

[6] E. Dahlhaus, Efficient parallel recognition algorithms of cographs and distance hereditary graphs, *Discrete Appl. Math.* 57 (1995) 29-44.

[7] A. Edenbrandt, Chordal graph recognition is in NC, *Inform. Process. Lett.* 24 (1987) 239-241.

[8] S. Foldes and P.L. Hammer, Split graphs, Proc. 8th Southeastern Conf. on Combitatorics Graph Theory and Computing (1977) 311-315.

[9] D.R Fulkerson and O.A. Gross, Incidence matrices and interval graphs, *Pacific J. Math.* 15 (1965) 835-855.

[10] M.C. Golumbic, The complexity of comparability graph recognition and coloring, *Computing* 18 (1977) 199-208.

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

[12] P. C. Gilmore and A. J. Hoffman, A characterization of comparability graphs and of interval graphs, *Canad. J. Math.* 16 (1964) 539-548.

[13] F. Harary, *Graph Theory*, Addison-Wesley, Reading, 1969.

[14] F. Harary and S.D. Nikolopoulos, "On Complete Systems of Invariants for Small Graphs", *Inter. J. of Comput. Math.* (1997) to appear.

[15] D. Helmbold and E.W. Mayr, Applications of parallel algorithms to families of perfect graphs, *Computing Suppl.* 7 (1990) 93-107.

[16] P.B. Henderson and Y. Zalcstein, A graph-theoretic characterization of the PV chunk class of synchronizing primitives, *SIAM J. Comput.* 6 (1977) 88-108.

[17] E. Howorka, A characterization of distance-hereditary graphs, *Quart. J. Math. Oxford Ser. 2* 28 (1977) 417-420.

[18] E. Howorka, A characterization of ptolemaic graphs, *J. of Graph Theory* 5 (1981) 323-331.

[19] C-W. Ho and R.C.T. Lee, Counting clique trees and computing perfect elimination schemes in parallel, *Inform. Process. Lett.* 31 (1989) 61-68.

[20] P.N Klein, Efficient parallel algorithms for chordal graphs, Proc. FOCS'89: 29th Symp. Fount. of Comp. Sci. (1989) 150-161.

[21] H. Lerchs, On cliques and kernels, Department of Computer Science (1971).

[22] R. Lin and S. Olariu, An NC recognition algorithm for cographs, *J. of Parallel and Distrib. Comput.* 13 (1991) 76-90.

[23] M. Murlet and J.P. Uhry, Parity graphs, in *Topics on Perfect Graphs*, C. Berge and V. Chvátal, eds, *Ann. Discrete Math. 21*, North Holland, Amsterdam, 1984.

[24] J. Naor, M. Naor and A. Schaffer, Fast parallel algorithms for chordal graphs, *19th Ann. ACM Symp. on Theory of Computing* (1987) 355-364.

[25] S.D. Nikolopoulos, Constant-time parallel recognition of split graphs, *Inform. Process. Lett.* 54 (1995) 1-8.

[26] S.D. Nikolopoulos and S.D. Danielopoulos, Parallel computation of perfect elimination schemes using partition techniques on triangulated graphs, *Computers and Mathematics with Applications* 29 (1995) 47-57.

[27] S.D. Nikolopoulos, Cographs and threshold graphs can be recognized in $O(1)$ time by a CRCW PRAM, Tech. Report TR-96-06, Department of Computer Science, University of Cyprus, 1996.

[28] A. Pnueli, A. Lempel and S. Even, Transitive orientation of graphs and identification of permutation graphs, *Canad. J. Math.* 23 (1971) 160-175.

[29] J. Reif (editor), *Synthesis of Parallel Algorithms*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1993.

[30] D.J. Rose, E. Tarjan and G.S. Lueker, Algorithmic aspects of vertex elimination on graphs, *SIAM J. Comput.* 5 (1976) 266-283.

[31] S. Seinsche, On a property of the class of n-colorable graphs, *J. of Comb. Theory: Series B*, 16 (1974) 191-193.

[32] R.E. Tarjan and M. Yannakakis, Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs, *SIAM J. Comput.* 13 (1984) 566-579.

[33] W.D. Wallis and J. Wu, Squares, clique graphs and chordality, *J. of Graph Theory* 20 (1995) 37-45.

[34] E.S. Wolk, The comparability graph of a tree, Proc. AMS 13 (1965) 789-795.