A Polynomial Algorithm for Recognizing Perfect Graphs

Gérard Cornuéjols [1]
GSIA, Carnegie Mellon University
Pittsburgh, PA 15213
gc0v@andrew.cmu.edu

Xinming Liu
GSIA, Carnegie Mellon University
Pittsburgh, PA 15213
xinming@andrew.cmu.edu

Kristina Vušković [2]
School of Computing
University of Leeds
Leeds LS2 9JT, UK
vuskovi@comp.leeds.ac.uk

November 15, 2002; revised February 6, 2003

[1]Supported by NSF grant DMI-0098427 and ONR grant N00014-97-1-0196.
[2]Supported by EPSRC grant GR/R35629/01.

Abstract

We present a polynomial algorithm for recognizing perfect graphs. This algorithm builds on a companion paper by Chudnovsky, Cornuéjols, Liu, Seymour and Vušković in the same issue of this journal and a decomposition theorem of Conforti, Cornuéjols and Vušković.

Keywords: perfect graph, recognition algorithm, cleaning, decomposition.

1 Introduction

All graphs in this paper are finite and simple. We follow the notation in West [17]. A graph G is *perfect* if, for every induced subgraph H of G, the chromatic number of H is equal to the size of a largest clique in H. In this paper we give a polynomial algorithm for recognizing whether a graph G is perfect. In the same issue, Chudnovsky and Seymour [9] present another polynomial algorithm for this problem. Both algorithms build on the companion paper by Chudnovsky, Cornuéjols, Liu, Seymour and Vušković [7], also published in the same issue. Previously, polynomial recognition algorithms for several subclasses of perfect graphs were known (Burlet and Fonlupt [3], Chvátal and Sbihi [5], Reed and Sbihi [16], Chvátal, Fonlupt, Sun and Zemirline [4]).

A *hole* is a chordless cycle of length at least four. A hole is *odd* if it contains an odd number of edges. Otherwise it is *even*. A graph G *contains* a graph H if H is isomorphic to an induced subgraph of G. A graph is H*-free* if it does not contain H. The complement of a graph G is denoted by $\bar{G}$. Recently, Chudnovsky, Robertson, Seymour and Thomas [8] proved that a graph G is perfect if and only if neither G nor $\bar{G}$ contains an odd hole. This was the famous strong perfect graph conjecture proposed by Berge [1]. Therefore one can check whether a graph G is perfect by checking whether G or $\bar{G}$ contains an odd hole.

We do not know how to check in polynomial time whether a general graph G contains an odd hole, but we know how to do so for a special class of graphs called "clean" graphs. If H is an odd hole in a graph G, we say that $u \in V(G) \setminus V(H)$ is *major* for H if its neighbors in H do not lie in a 2-edge path of H. We say that H is *clean* if G contains no major vertex for H. A graph G is "clean" if, whenever G contains an odd hole then all odd holes in a certain family (to be defined later) are clean. Our approach to recognizing whether a graph G is perfect consists of two steps:

(i) constructing in polynomial time a clean graph G' that contains an odd hole if and only if G does, or in some cases stopping with the conclusion that G is imperfect, and

(ii) checking whether the clean graph G' contains an odd hole.

The difficult part of the cleaning step (i) is given in the companion paper [7]. The strategy of cleaning was first introduced by Conforti and Rao [14] to recognize linear balanced matrices. It was also used in recognition algorithms for balanced matrices [12], [10] and for even holes in graphs [11].

The main result of this paper is a polynomial algorithm for step (ii): checking whether a clean graph contains an odd hole. Our approach is by decomposition. Chudnovsky and Seymour [9] have a different polynomial algorithm for step (ii), which does not use decomposition. The problem of checking whether a general graph contains an odd hole is still open. Interestingly, it is NP-complete to recognize whether a graph contains an odd hole passing through a specific vertex (Bienstock [2]).

1.1 Decomposition Theorem

The idea of our algorithm is to decompose the input graph G into a polynomial number of simpler graphs $G_1, \dots, G_m$ so that the following two properties are satisfied:

(1) G is odd-hole-free if and only if G_i is odd-hole-free for every $i = 1, \dots, m$, and

(2) for every $i = 1, \dots, m$ it is easy to check directly whether G_i is odd-hole-free.

In order to satisfy (1), clean graphs will be very helpful. We use the following decomposition theorem for odd-hole-free graphs by Conforti, Cornuéjols and Vušković [13] as the basis of our recognition algorithm.

A set S of vertices is a *double star* if S contains two adjacent vertices u and v such that $S \subseteq N(u) \cup N(v)$. Here $N(x)$ denotes the set of vertices adjacent to vertex x. We say that S is *centered at uv*. The vertex set S is a *cutset* of G if $G \setminus S$ contains more connected components than G.

A graph G has a *2-join* $V_1|V_2$ with special sets (A_1, A_2, B_1, B_2) if its vertices can be partitioned into sets V_1 and V_2 so that, for $i = 1, 2$, V_i contains disjoint, nonempty vertex sets A_i and B_i, such that every vertex of A_1 is adjacent to every vertex of A_2, every vertex of B_1 is adjacent to every vertex of B_2, and there are no other adjacencies between V_1 and V_2. Furthermore, for $i = 1, 2$, $|V_i| > 2$ and the graph induced by V_i is not a chordless path. 2-Joins were introduced by Cornuéjols and Cunningham [15].

A *basic* graph is a bipartite graph or the line graph of a bipartite graph or the complement of a line graph of a bipartite graph.

1.1 *[13] If G is an odd-hole-free graph, then either G is basic, or G has a double star cutset or a 2-join.*

Checking whether a graph is basic can easily be done in polynomial time. This is an answer to (2) above. A polynomial algorithm for finding a 2-join is given in [15] and [11]. It is not difficult to find a double star cutset in polynomial time: Let u, v be two adjacent vertices and x, y two nonadjacent vertices. We can test whether there is a double star cutset centered at u and v that disconnects x and y by removing all the neighbors of u and all the neighbors of v except x and y. Therefore the main difficulty in applying Theorem 1.1 is to decompose a graph G that has a double star cutset or a 2-join into "blocks of decomposition" G_i that satisfy (1) above.

Instead of Theorem 1.1, one could use the decomposition theorem of Chudnovsky, Robertson, Seymour and Thomas [8] (or a strengthening due to Chudnovsky [6]). Note however that 2-joins are defined slightly differently in [8], [6] (the graph induced by V_i may be a chordless path), which poses a difficulty.

1.2 Blocks of Decomposition

We now define the blocks of decomposition for double star cutsets and 2-joins. Remember that our goal is to satisfy (1) above.

2-Join Decomposition: Let $V_1|V_2$ be a 2-join of G with special sets (A_1, A_2, B_1, B_2). If there does not exist a path from a vertex of A_2 to a vertex of B_2 in $G[V_2]$ then we define *block* G_1 to be the subgraph of G induced by $V_1 \cup \{a_2, b_2\}$, where $a_2 \in A_2$ and $b_2 \in B_2$. Otherwise, let Q_2 be a shortest path from A_2 to B_2 in $G[V_2]$. We define *block* G_1 to be the graph obtained from $G[V_1 \cup V(Q_2)]$ by replacing Q_2 by a chordless path P_2 of length 4 if Q_2 is of even length, and of length 5 otherwise. Path P_2 is called the *marker path.* Block G_2 is defined similarly.

1.2 *Let G_1 and G_2 be the blocks of a 2-join decomposition of G. Then G is odd-hole-free if and only if G_1 and G_2 are odd-hole-free. Furthermore, if G contains a clean odd hole of length greater than 5, then G_1 or G_2 contains a clean odd hole of length greater than 5.*

Proof. Assume first that G is odd-hole-free and that G_1 contains an odd hole H. Suppose that H does not contain the marker path P_2. Since H is not a hole of G, it contains both endvertices of Q_2 and Q_2 is an edge. But then G contains a shorter odd hole, a contradiction. Therefore H contains the marker path P_2. Let $H_1 = H \cap V_1$. Then $H_1 \cup Q_2$ induces an odd hole of G, a contradiction.

Next we prove that if G contains an odd hole H, then G_1 or G_2 contains an odd hole. If H does not contain a vertex in each of A_1, A_2, B_1, B_2, say A_2, then H is contained in G_1. Thus we can assume that H contains a vertex in each of A_1, A_2, B_1, B_2. First assume that H contains exactly four vertices of $A_1 \cup A_2 \cup B_1 \cup B_2$. Let $H_1 = H \cap V_1$ and $H_2 = H \cap V_2$. Since H is odd, w.l.o.g. H_1 is even and H_2 is odd. If P_2 is odd then $H_1 \cup P_2$ is an odd hole of G_1. Hence we can assume that P_2 is even. Then one of $P_1 \cup Q_2$ or $P_1 \cup H_2$ induces an odd hole in G_2. Now assume that H contains more than four vertices of $A_1 \cup A_2 \cup B_1 \cup B_2$. W.l.o.g. H contains two vertices $a_1, a_1' \in A_1$. Then H contains exactly one vertex in A_2, which is adjacent to a_1 and a_1'. Since H is a hole it cannot contain more than one vertex in B_2. Hence H is entirely contained in G_1.

The second statement of the theorem follows by observing that if the odd hole H defined in the previous paragraph is a clean odd hole of length greater than 5, then the odd hole found in G_1 or G_2 is also a clean odd hole of length greater than 5 (for example the fact that $P_1 \cup Q_2$ is clean follows from the fact that Q_2 is a shortest path from A_2 to B_2 in $G[V_2]$). ∎

Double Star Decomposition: Let S be a double star cutset of G and $C_1, C_2, \dots, C_n$ the connected components of $G \setminus S$. We define the blocks of decomposition to be the graphs $G_1, \dots, G_n$ where $G_i = G[V(C_i) \cup S]$.

This definition of blocks for the double star cutset does not preserve the odd-hole-free property. Consider a graph G that consists of a 5-hole $H = x_1, x_2, x_3, x_4, x_5, x_1$ and a vertex x adjacent to x_1, x_2 and x_4. If we decompose G with a double star cutset $N(x) \cup \{x\}$ then neither of the blocks contains an odd hole. To overcome this difficulty, we first clean G before performing double star decompositions. In Section 3.1, we show that double star decompositions preserve the odd-hole-free property when G is "clean" as defined below.

1.3 Cleaning

Cleaning the input graph is key to our algorithm. Next we give the relevant definitions.

Let H be an odd hole in a graph G. Recall that $u \in V(G) \setminus V(H)$ is *major* for H if its neighbors in H do not lie in a 2-edge path of H. A major vertex is a *racket* if it has exactly three neighbors in H, two of which are adjacent. A subset $X \in V(G)$ is a *cleaner* for H if it contains every major vertex for H and $X \cap V(H)$ is contained in a 2-edge path of H. The following result is obtained in [7].

1.3 *There is a $O(|V(G)|^6)$ algorithm with the following specifications:*

- **Input** *A graph G.*
- **Output** *Either*
  - *(i) A family $\mathcal{X}$ of $O(|V(G)|^5)$ subsets, such that if H is a shortest odd hole in G with no racket, then some $X \in \mathcal{X}$ is a cleaner for H, or*
  - *(ii) it determines that G is IMPERFECT.*

A cleaner $X \in \mathcal{X}$ in 1.3 is obtained when H has no racket. In Section 2, we give a direct way of cleaning when H has a racket.

A vertex $u \in V(G) \setminus V(H)$ is *clean* for H if it is not major for H. For a clean vertex u, let P be a subpath of H of length at most two (P possibly empty) such that $N(u) \cap V(H) \subseteq V(P)$ and u is adjacent to both endvertices of P. We say that u is a clean vertex of Type *ci* for H if $i = |V(P)|$.

Let H be an odd hole and u a clean vertex of Type c3 for H, with neighbors in H contained in a subpath u_1, u_2, u_3 of H. Let H' be the hole induced by $(V(H) \setminus \{u_2\}) \cup \{u\}$. We say that H' is obtained from H through a *vertex substitution.* Note that H and H' have the same length.

A *tent* for an odd hole H is an edge uv such that both u and v are clean vertices for H, and for some $u'v'$-subpath P of H of length three, $(N(u) \cup N(v)) \cap V(H) \subseteq V(P)$, and u is adjacent to u', and v is adjacent to v'. Note that u is not adjacent to v' and v is not adjacent to u'. Let H' be the hole induced by $(V(H) \setminus V(P)) \cup \{u, v, u', v'\}$. We say that H' is obtained from H through a *tent substitution.* Note that H and H' have the same length.

Let H be an odd hole in a graph G. We define $\mathcal{C}_G(H)$ to be the family of all holes of G obtained from H through a sequence of vertex substitutions or tent substitutions. We say that H is *spotless* if every hole of $\mathcal{C}_G(H)$ is clean.

A graph G is *clean* if it is either odd-hole-free or it contains a spotless shortest odd hole.

2 Constructing a Clean Graph

Building on 1.3, the following theorem will be proved in this section.

2.1 *There is a $O(|V(G)|^{10})$ algorithm with the following specifications:*

- **Input** *A graph G.*
- **Output** *Either*
  - *(i) A family $\mathcal{F}$ of $O(|V(G)|^{10})$ induced subgraphs of G, such that $\mathcal{F}$ contains a graph F with a spotless odd hole if and only if G contains an odd hole, or*
  - *(ii) it determines that G is IMPERFECT.*

In order to prove this, we first check in Section 2.1 whether G contains two types of easily detectable structures that necessarily contain an odd hole. By 1.3, a cleaner can be obtained if a shortest odd hole H^* has no racket in G. In Section 2.2, we give a direct way of cleaning if H^* has a racket. Finally, we present our cleaning algorithm in Section 2.3, which implies Theorem 2.1.

Throughout this section, unless otherwise stated, we assume that H^* is a shortest odd hole of G.

2.1 Two Easily Detectable Structures

A *wheel*, denoted by (H, x), is a graph induced by a hole H and a vertex $x \notin V(H)$ having at least three neighbors in H, say $x_1, \ldots, x_n$. Vertex x is the *center* of the wheel. A subpath of H connecting x_i and x_j is a *sector* if it contains no intermediate vertex x_l, $l \in \{1, \ldots, n\}$. A *short sector* is a sector of length 1, and a *long sector* is a sector of length at least 2. A wheel is *odd* if it contains an odd number of short sectors, and *even* otherwise. Each of the long sectors together with vertex x induces a hole. If each of these holes is even and the wheel (H, v) is odd then H is an odd hole, since the wheel (H, x) contains an odd number of short sectors. Therefore, every odd wheel contains an odd hole. An odd wheel (H, u) is *short* if u has 4 neighbors in H and the wheel has exactly one long sector.

A *3-path configuration* (*3PC*), denoted by $3PC(xyz; u)$, is a graph induced by three chordless paths $P_1 = x, \ldots, u$, $P_2 = y, \ldots, u$ and $P_3 = z, \ldots, u$ having no common or adjacent intermediate vertices, such that at most one of the paths is of length 1 and the vertex set $\{x, y, z\}$ induces a clique of size 3. Every two of the paths P_1, P_2, P_3 induce a hole. Since two of the three paths must have the same parity, one of these holes is odd. Therefore, every 3-path configuration contains an odd hole. A 3PC is *short* if P_1 is of length one and P_2 is of length at most four.

Short odd wheels and short 3PC's can be detected in $O(|V(G)|^9)$ time.

2.2 Technical Lemmas

A family $\mathcal{L}$ of holes of G is said to be *racket-free* if no vertex of G is a racket for any of the holes in $\mathcal{L}$.

2.2 *If $u \in V(G) \setminus V(H^*)$ has a neighbor but no more than three neighbors in H^*, then u is a racket or a clean vertex of Type c1, c2 or c3 for H^*.*

Proof. If u has one neighbor in H^* then u is a clean vertex of Type c1 for H^*. Now suppose that u has two neighbors in H^*, say u_1 and u_2. Let P_1 and P_2 be the two u_1u_2-subpaths of H^*. Since H^* is odd, P_1 and P_2 have different parity, say P_1 is odd. If P_1 is of length 1 then u is a clean vertex of Type c2 for H^*. Otherwise, $V(P_1) \cup \{u\}$ induces an odd hole. Since this hole cannot be shorter than H^*, P_2 is of length 2, and hence u is a clean vertex of Type c3 for H^*.

Now assume that u has three neighbors in H^*, and let P_1, P_2 and P_3 be the three sectors of the wheel (H^*, u). If exactly one of the sectors is short then u is a racket for H^*. If two of the sectors are short then u is of Type c3 for H^*. Finally suppose that all three sectors are long. Since H^* is odd, at least one of the sectors, say P_1, is odd. Then $V(P_1) \cup \{u\}$ induces an odd hole shorter than H^*, a contradiction. ∎

2.3 *Suppose that a graph G contains a shortest odd hole H^* but not a short 3PC. Let u be a racket for H^*, and let $P_1 = x_0, x_1, x_2, x_3$ and $P_2 = y_0, y_1, y_2, y_3, y_4$ be two subpaths of H^* such that u is adjacent to x_1, x_2 and y_2. Let $G' = G \setminus [(\cup_{i=1}^{2} N(x_i) \cup \cup_{i=1}^{3} N(y_i) \cup N(u)) \setminus (V(P_1) \cup V(P_2))]$. Then*

- *(i) If v is a major vertex for H^* then v has a neighbor in $\{x_1, x_2, y_1, y_2, y_3, u\}$.*
- *(ii) G' contains H^* and H^* is spotless in G'.*
- *(iii) Any hole of $\mathcal{C}_{G'}(H^*)$ has a pair of vertices at distance at least four in G'.*

Proof. We first prove (i). Let v be a major vertex for H^*, and suppose that v is not adjacent to any vertex in $\{x_1, x_2, y_1, y_2, y_3, u\}$. Let Q_1 be the x_1y_2-subpath of H^* that does not contain x_2, and let Q_2 be the x_2y_2-subpath of H^* that does not contain x_1. Since H^* is a shortest odd hole, (H^*, v) has an odd number of short sectors. Hence, we may assume w.l.o.g. that Q_2 contains an odd number of short sectors of v. If v has at least three neighbors in Q_2 then $V(Q_2) \cup \{u, v\}$ induces an odd wheel centered at v. This odd wheel contains an odd hole shorter that H^*, a contradiction. Therefore, v

has exactly two neighbors in Q_2, denoted by v_1 and v_2, and v_1 is adjacent to v_2. Let v_3 be a vertex of Q_1 that is adjacent to v such that the v_3y_2-subpath Q_1' of Q_1 does not contain any other neighbor of v. Then $V(Q_1') \cup V(Q_2) \cup \{u, v\}$ induce a $3PC(v_1v_2v; y_2)$ which contains an odd hole shorter than H^*.

We claim that any hole of $\mathcal{C}_{G'}(H^*)$ contains all vertices in $\{x_1, x_2, y_1, y_2, y_3\}$. If H does not contain x_1 or x_2 then by the construction of G', the graph G contains a tent z_1z_2 for H^* such that z_1 is adjacent to x_0, z_2 is adjacent to x_3 and neither z_1 nor z_2 is adjacent x_1 or x_2. This implies that $G[V(H^*) \cup \{u, z_1, z_2\}]$ contains a short 3PC, a contradiction. Hence, H contains both x_1 and x_2. Then by construction of G' and $\mathcal{C}_{G'}(H^*)$, H contains all the vertices of $\{x_1, x_2, y_1, y_2, y_3\}$.

Now we prove (ii). Obviously, G' contains H^*. Suppose that some hole H of $\mathcal{C}_{G'}(H^*)$ is not clean. Let $v \in V(G')$ be a major vertex for H in G'. Note that H is also a shortest odd hole in G. In G, vertex u is a racket for H adjacent to x_1, x_2 and y_2 since if u has another neighbor z in H then G' would contain $z \in N(u) \setminus (V(P_1) \cup V(P_2))$, a contradiction. Then by (i), v has a neighbor in $\{x_1, x_2, y_1, y_2, y_3, u\}$, another contradiction.

Finally, we prove (iii). Since G contains no short 3PC, the long sectors of (H^*, u) are of length greater than four. So x_1 and y_2 are at distance at least four in H^*. Since x_1 and y_2 have degree two in G', the neighbors of x_1 and y_2 are not adjacent in G'. Therefore, x_1 and y_2 are at distance at least 4 in G'. ∎

Given an odd hole H, a *long racket* for H is an edge uv such that u is a clean vertex of Type c2 for H, v is a clean vertex of Type c1 for H, and each subpath of H between the neighbors of u and the neighbor of v has length at least three. A family $\mathcal{L}$ of holes of G is said to be *long-racket-free* if no edge of G is a long racket for any of the holes in $\mathcal{L}$.

2.4 *Suppose that a graph G contains a shortest odd hole H^* but not a 5-hole, a 7-hole or a short 3PC. Suppose also that there is no racket for H^* in G. Let uv be a long racket for H^*, and let $P_1 = x_0, x_1, x_2, x_3$ and $P_2 = y_0, y_1, y_2, y_3, y_4$ be two subpaths of H^* such that u is adjacent to x_1, x_2 and v is adjacent to y_2. Let $G' = G \setminus [(\cup_{i=1}^{2} N(x_i) \cup \cup_{i=1}^{3} N(y_i) \cup N(u)) \setminus (V(P_1) \cup V(P_2))]$. Then*

(i) If v' is a major vertex for H^* then v' has a neighbor in $\{x_1, x_2, y_1, y_2, y_3, u\}$.

(ii) G' contains H^* and H^* is clean in G'.

(iii) Any hole of $\mathcal{C}_{G'}(H^*)$ has a pair of vertices at distance at least four in G'.

Proof. We first prove (i). Let v' be a major vertex for H^* but not a racket, and suppose that v' is not adjacent to any vertex in $\{x_1, x_2, y_1, y_2, y_3, u\}$. Let Q_1 be the x_1y_2-subpath of H^* that does not contain x_2, and let Q_2 be the x_2y_2-subpath of H^* that does not contain x_1. Since H^* is a shortest odd hole, the wheel (H^*, v') has an odd number of short sectors. Hence, we may assume w.l.o.g. that Q_2 contains an odd number of short sectors of v'. If v' has at least three neighbors in $V(Q_2) \cup \{v\}$ then $V(Q_2) \cup \{u, v, v'\}$ induces an odd wheel centered at v'. This odd wheel contains an odd hole shorter that H^*, a contradiction. Therefore, v' is not adjacent to v and has exactly two neighbors in Q_2, denoted by v_1' and v_2', and v_1' is adjacent to v_2'. Let v_3' be a vertex of Q_1 that is adjacent to v' such that the $v_3'y_2$-subpath Q_1' of Q_1 does not contain any other neighbor of v'. Note that by our assumption v' is not a racket for H^* and hence it has at least four neighbors in H^*. Then $V(Q_1') \cup V(Q_2) \cup \{u, v, v'\}$ induce a $3PC(v_1'v_2'v'; y_2)$ which contains an odd hole shorter than H^*.

Now (ii) follows from the construction of G'.

We claim that any hole of $\mathcal{C}_{G'}(H^*)$ contains all vertices in $\{x_1, x_2, y_1, y_2, y_3\}$. If H does not contain x_1 or x_2 then by the construction of G', the graph G contains a tent z_1z_2 for H^* such that z_1 is adjacent to x_0, z_2 is adjacent to x_3 and neither z_1 nor z_2 is adjacent x_1 or x_2. This implies that $G[V(H^*) \cup \{u, v, z_1, z_2\}]$ contains a short 3PC, a contradiction. Hence, H contains both x_1 and x_2. Then by construction of G' and $\mathcal{C}_{G'}(H^*)$, H contains all the vertices of $\{x_1, x_2, y_1, y_2, y_3\}$.

Since uv is a long racket for H^*, and H^* is of length greater than seven, we may assume w.l.o.g. that x_1 and y_2 are at distance at least four in H^*. Now (iii) holds by the same argument as in the proof of (iii) of 2.3. ∎

2.5 *Let G be a graph containing a shortest odd hole H^* but no 5-hole, no 7-hole, no short odd wheel and no short 3PC. If H^* is clean and $\mathcal{C}_G(H^*)$ is racket-free then H^* is spotless.*

Proof. The proof follows from the following two claims.

Claim 1: Any hole obtained from H^* through a vertex substitution is clean.

Proof of Claim 1: Let x be a Type c3 vertex for H^*. Let x_1, x_2, x_3 be the subpath of H^* such that x is adjacent to x_1 and x_3. Let H be the odd hole obtained from H^* by substituting x for x_2. Suppose u is a major vertex for H. Since u is a clean vertex for H^*, u must be adjacent to x. If u is of Type c1 for H^* then the subgraph induced by $V(H) \cup \{u\}$ contains an odd hole shorter than H^*. If u is a vertex of Type c2 for H^* then u is a racket for H, contradicting the fact that $\mathcal{C}_G(H^*)$ is racket-free. So u is of Type c3 for H^*. But then the subgraph induced by $V(H) \cup \{u\}$ contains an odd hole shorter than H^*, or a short wheel or a short 3PC (if u is adjacent to x_1 or x_3). This completes the proof of Claim 1.

Claim 2: Any hole obtained from H^* through a tent substitution is clean.

Proof of Claim 2: Let u_1u_2 be a tent for H^*. Let y_1, y_2, y_3, y_4 be the subpath of H^* such that $(N(u_1) \cup N(u_2)) \cap V(H^*) \subseteq \{y_1, y_2, y_3, y_4\}$, and u_1 is adjacent to y_1, and u_2 is adjacent to y_4. Note that u_1 is not adjacent to y_4 and u_2 is not adjacent to y_1. By Claim 1 and since G does not contain a 5-hole, we may assume that u_1 and u_2 are either both of Type c1 or both of Type c2. Let H be the hole induced by $V(H^*) \cup \{u_1, u_2\} \setminus \{y_2, y_3\}$. Suppose u is a major vertex for H. Since u is a clean vertex for H^*, u must be adjacent to u_1 or u_2. Consider the following cases.

Case 1 u is adjacent to exactly one of u_1 or u_2, say u_1.

Let P be a shortest subpath of H^* that contains all the neighbors of u and u_1 in H^*. Since u is major for H, P is of length at least 2. Suppose P is of length 2. Since u is major for H, u_1 must be of Type c1 for H^*. If u is of Type c1 for H^*, then there is a 5-hole. If u is of Type c2 for H^*, then u is a racket for H. So u is of Type c3 for H^*. But then either u is a racket for H or there is a short odd wheel.

Now suppose that P is of length 3. Let z_1, z_2, z_3, y_1 be the subpath of H^* not containing y_2. Since u is major for H and it cannot be a racket for H, u must be adjacent to z_1 or z_2. If u is adjacent to z_1 then $(V(H^*) \cup \{u_1, u_2, u\}) \setminus \{z_2, z_3, y_1, y_2, y_3\}$ induces an odd hole shorter than H^*. So u is not adjacent to z_1, and hence it is adjacent to z_2. If u is adjacent to neither y_1 nor z_3, then there is a 5-hole. If u is adjacent to both y_1 and z_3, then there is a short odd wheel. Otherwise, u is a racket for H.

Therefore, P is of length greater than 3. Suppose that u is of Type c3 for H^*. Let H' be the hole obtained by substituting u into H^*. Then by 2.2, u_1 is a racket for H'. If u is of Type c2 for H^*, then u is a racket for H. So u is of Type c1 for H^*. But then there is an odd hole shorter than H^*.

Case 2 u is adjacent to both u_1 and u_2.

If u is of Type c1 for H^* then u is a racket for H^*. Suppose u is of Type c2 for H^*. If u is adjacent to y_1 or y_4, then there is a short odd wheel. Else $G[V(H) \cup \{u\}]$ contains an odd hole shorter than H^*. So u is of Type c3 for H^*. If u is adjacent to y_2 or y_3, then either u is a racket for H or there is a short odd wheel. If u is adjacent to y_1 or y_4 then $G[V(H) \cup \{u\}]$ contains an odd hole shorter than H^*. Otherwise, let H' be the hole obtained by substituting u into H^*. By Claim 1 applied to H', u_1 and u_2 must both be of Type c3 for H'. But then H^* is of length 7, a contradiction. ∎

2.6 *Let G be a graph that contains a shortest odd hole H^* such that $\mathcal{C}_G(H^*)$ is racket-free. Suppose that G contains no 5-hole, no short odd wheel and no short 3PC. If $u, v \in V(G) \setminus V(H^*)$ are two adjacent clean vertices for H^* then one of the following is true.*

(i) uv is a tent or a long racket for H^*.

(ii) The vertices of $(N(u) \cup N(v)) \cap V(H^*)$ are contained in a subpath P of H^* of length at most two, and if P is of length 2 then u or v is a clean vertex of Type c3 for H^*.

Proof. Let P be a shortest subpath of H^* such that the vertices of $(N(u) \cup N(v)) \cap V(H^*)$ are contained in P. Suppose that P is of length 2. Vertices u and v cannot be both of Type c1 since otherwise, G contains a 5-hole. If neither u nor v is of Type c3 for H^*, $V(H^*) \cup \{u, v\}$ induces a short odd wheel or a short 3PC. Therefore, (ii) holds. If P is of length 3 then (i) holds. Now we assume that P is of length greater than 3.

Suppose that one of u and v is clean vertex of Type c3 and the other of Type c_2, say u is of Type c3. Let H be a hole obtained by substituting u into H^*. Then v is a racket for H, contradicting the assumption that $\mathcal{C}_G(H^*)$ is racket-free. If one of u, v is of Type c2 and the other of Type c1, then uv is a long racket for H^*. If u and v are both of Type c3 and they have a common neighbor in H^* then $G[V(H^*) \cup \{u, v\}]$ contains a short odd wheel or a short 3PC. In all other cases, $G[V(H^*) \cup \{u, v\}]$ contains an odd hole shorter than H^*. ∎

2.3 Algorithm for Constructing a Clean Graph

Now we present our algorithm for constructing a clean graph as follows.

Input: A graph G.

Output: IMPERFECT if G is not perfect or a family $\mathcal{F}$ of induced subgraphs of G that satisfies the following properties:

(1) G is odd-hole-free if and only if all graphs of $\mathcal{F}$ are odd-hole-free. Furthermore, if G contains a shortest odd hole H^* then for some graph $F \in \mathcal{F}$, F contains H^* and H^* is spotless in F, and any odd hole of $\mathcal{C}_F(H^*)$ has a pair of vertices at distance at least four in F.

(2) $|\mathcal{F}|$ is $O(|V(G)|^{10})$.

Step 1: Check whether G contains a 5-hole, a 7-hole, a short odd wheel or a short 3PC. If it does output IMPERFECT and stop. Otherwise, set $\mathcal{F} = \{G\}$.

Step 2: For every (P_1, P_2, u) where $P_1 = x_0, x_1, x_2, x_3$ and $P_2 = y_0, y_1, y_2, y_3, y_4$ are two chordless paths, and $u \in N(x_1) \cap N(x_2)$, add to $\mathcal{F}$ the graph obtained from G by removing the vertex set $(\cup_{i=1}^{2} N(x_i) \cup \cup_{i=1}^{3} N(y_i) \cup N(u)) \setminus (V(P_1) \cup V(P_2))$.

Step 3: Apply Algorithm 1.3 to G. If the output is IMPERFECT return the same and stop. Otherwise, Let $\mathcal{X}$ denote the output family of the algorithm. For each $X \in \mathcal{X}$ and each $Y \subseteq X$ obtained by removing from X at most three vertices that are contained in a 2-edge path, add the graph $G \setminus Y$ to $\mathcal{F}$.

2.7 *This algorithm produces the desired output, and its running time is* $O(|V(G)|^{10})$.

Proof. Suppose that the algorithm does not output IMPERFECT. Suppose G contains a shortest odd hole H^*. Note that holes of $\mathcal{C}_G(H^*)$ are of length greater than 7. If $\mathcal{C}_G(H^*)$ is not racket-free then (1) holds by 2.3 and Step 2. If $\mathcal{C}_G(H^*)$ is racket-free but not long-racket-free then (1) holds by 2.4 and 2.5 and Step 2. Finally suppose that $\mathcal{C}_G(H^*)$ is both racket-free and long-racket-free. By 1.3 and 2.5 and Step 3 it follows that for some graph $F \in \mathcal{F}$, F contains H^* and H^* is spotless in F. Let H be any hole in $\mathcal{C}_F(H^*)$. Since H is of length greater than 7, it contains two vertices x and y such that they are at distance at least 4 in H. Suppose there is a path P from x to y in F of length less than 4. Since H is clean, P cannot be of length 2. So P is of length 3. But then by 2.6 there is a long racket for H, contradicting the assumption that $\mathcal{C}_G(H^*)$ is long-racket-free. Therefore, x and y are at distance at least 4 in F, and hence (1) holds.

$O(|V(G)|^{10})$ graphs are created in Step 2, and $O(|V(G)|^{8})$ graphs are created in Step 3. Hence, (2) holds.

The running time of Step 2 is $O(|V(G)|^{10})$: Indeed, for each graph created in Step 2 we only need to keep a constant amount of information, namely the 10 vertices in $V(P_1) \cup V(P_2) \cup \{u\}$. The running time of Step 1 and 3 is $O(|V(G)|^{9})$. Therefore, the overall running time is $O(|V(G)|^{10})$. ∎

3 Decomposing Clean Graphs

We first perform double star cutset decompositions recursively and, when the resulting blocks contain no more double star cutsets, we perform 2-join decompositions without creating new double star cutsets, until no 2-joins exist either. We will show that, when G is clean, G is odd-hole-free if and only if all the blocks are odd-hole-free. Therefore, in order to check whether G is odd-hole-free, it suffices to check whether all the blocks are basic, by 1.1.

3.1 Double Star Decomposition

In this section we decompose a graph using double star decompositions.

Let uv be a long racket for an odd hole H, and let $P_1 = x_0, x_1, x_2, x_3$ and $P_2 = y_1, y_2, y_3$ be two subpaths of H such that u is adjacent to x_1, x_2 and v is adjacent to y_2, and y_1 is on the x_1y_2-subpath of H that does not contain x_2. It is easy to see that a double star cutset S' centered at uv may destroy the odd hole H. In order to overcome this problem, we first check whether such a double star cutset S' that detroys H exists, as follows. Note that if S' destroys H then $N(u) \cup N(v)$ is also a double star cutset that destroys H. Let Q_1 be the x_0y_1-subpath of H that does not contain x_1, and Q_2 the x_3y_3-subpaths of H that does not contain x_2. We say that uv is a *detectable* long racket for H if $S = N(u) \cup N(v)$ is a double star cutset centered at uv such that Q_1 and Q_2 are contained in different connected components of $G \setminus S$. If uv is a detectable long racket for H then there is a 3PC$(x_1x_2u; y_2)$ such that one of the three paths is y_2, v, u and the intermediate vertices of the other two paths are contained in two different connected components of $G \setminus S$. Such a 3PC$(x_1x_2u; y_2)$ can be detected in $O(|V(G)|^7)$ time by enumerating all possible sets $\{x_0, x_1, x_2, x_3, y_2\}$ and then using a connectivity algorithm. In this case, G contains an odd hole.

3.1 *Let G be a graph that does not contain a 5-hole, a short odd wheel or a short 3PC, and contains a spotless shortest odd hole H^*. If G is decomposed by a double star cutset S centered at uv, then either uv is a detectable long racket for some odd hole or some hole $H \in \mathcal{C}_G(H^*)$ is entirely contained in one of the blocks of decomposition.*

Proof. Suppose that uv is not a detectable long racket for any odd hole. Assume that H^* is broken by S, i.e. H^* is not contained in any block of decomposition. Then H^* contains a vertex of S, but it does not contain both u and v. Suppose that H^* contains u, but not v. Since H^* is broken by S, v is clean vertex of Type c3 for H^*. Hence the hole obtained by substituting v into H^* is in $\mathcal{C}_G(H^*)$ and entirely contained in one of the blocks of decomposition. So we may assume that H^* contains neither u nor v. Then by 2.6, one of the holes in $\mathcal{C}_G(H^*)$ is entirely contained in one of the blocks of decomposition. ∎

Double Star Decomposition Algorithm

Input: A clean graph G that does not contain a short odd wheel, short 3PC, 5-hole or 7-hole, and that contains a spotless odd hole H^* such that every hole of $\mathcal{C}_G(H^*)$ contains a pair of vertices at distance at least four if G contains an odd hole.

Output: Either an odd hole of G, or a family $\mathcal{L}$ of induced subgraphs of G that satisfies the following properties:

(1) G is odd-hole-free if and only if all graphs in $\mathcal{L}$ are odd-hole-free.

(2) The graphs in $\mathcal{L}$ do not have a double star cutset.

(3) The number of graphs in $\mathcal{L}$ is $O(|V(G)|^2)$.

Step 1: Set $\mathcal{L} = \emptyset$ and $\mathcal{L}' = \{G\}$.

Step 2: If $\mathcal{L}' = \emptyset$ then stop. Otherwise, remove a graph F from $\mathcal{L}'$. If the distance of every pair of vertices of F is less than 4 in G, go to Step 2. If F has no double star cutset, then add F to $\mathcal{L}$ and go to Step 2. Otherwise, let S be a double star cutset centered at uv in F. If uv is a detectable long racket then identify an odd hole and stop. Otherwise, construct the blocks of decomposition, add them to $\mathcal{L}'$ and go to Step 2.

3.2 *This algorithm produces the desired output, and its running time is $O(|V(G)|^9)$.*

Proof. (2) holds by the construction of the algorithm. We now show that (1) holds. Since the graphs in $\mathcal{L}$ are induced subgraphs of G, if G is odd-hole-free then all graphs in $\mathcal{L}$ are odd-hole-free. Suppose G contains a spotless shortest odd hole H^*. Recall that every hole in $\mathcal{C}_G(H^*)$ contains two vertices that are at distance at least 4 in G. Hence by 3.1, the output is an odd hole or some graph in $\mathcal{L}$ contains an odd hole of $\mathcal{C}_G(H^*)$.

We prove (3) by showing that the number of graphs in $\mathcal{L}$ is bounded by the number of pairs of vertices at distance at least 4 in G. Let S be a double star cutset of a graph F, and let $F_1, \dots, F_m$ be the blocks of decomposition. Let u and v be two vertices of F that are at distance at least 4 in G (and hence in F). Vertices u and v cannot be contained in two different blocks of decomposition since otherwise they would both have to be in S, but since S is a double star, all vertices of S are at distance at most 3. Therefore no pair of vertices that are at distance at least 4 in G can be contained in different graphs in $\mathcal{L}$.

The running time is determined by the complexity of finding dectectable long rackets, which is $O(|V(G)|^7)$, and the fact that the algorithm considers at most $O(|V(G)|^2)$ double star cutsets in Step 2. ∎

3.2 2-Join Decomposition

In this section we decompose a graph that has no double star cutset using 2-join decompositions, without creating any new double star cutset.

3.3 *If a graph G has a 2-join $V_1|V_2$ with special sets (A_1, A_2, B_1, B_2) such that $V_1 \setminus (A_1 \cup B_1) = \emptyset$ and $V_2 \setminus (A_2 \cup B_2) = \emptyset$ then G contains no clean odd hole of length greater than five.*

Proof. Suppose that H is a clean odd hole of length at least seven in G. Since H has no major vertex, H cannot be entirely contained in $A_1 \cup B_1$ or $A_2 \cup B_2$. Now we assume w.l.o.g. that H contains one vertex of A_2 and two vertices of A_1. Then H contains at least two vertices of B_1 but no vertex of B_2. Now any vertex of B_2 is major for H, a contradiction. ∎

3.4 *Suppose that a graph G has a 2-join $V_1|V_2$ with special sets (A_1, A_2, B_1, B_2) such that at least one of $V_1 \setminus (A_1 \cup B_1)$ and $V_2 \setminus (A_2 \cup B_2)$ is nonempty. Let G_1 and G_2 be the blocks of a 2-join decomposition of G. If G does not have a double star cutset then the following hold.*

(1) Both $V_1 \setminus (A_1 \cup B_1)$ and $V_2 \setminus (A_2 \cup B_2)$ are nonempty.

(2) For $i = 1, 2$, G_i does not have a double star cutset.

(3) For $i = 1, 2$, $|V_i| \geq 6$.

Proof. For $i = 1, 2$ let $\mathcal{P}_i$ be the set of all paths in $G[V_i]$ with one vertex in A_i, the other in B_i and no intermediate vertex in $A_i \cup B_i$. (1) follows from the following claim.

Claim 1: For $i = 1, 2$, $\mathcal{P}_i \neq \emptyset$ and all paths of $\mathcal{P}_i$ are of length greater than 1.

Proof of Claim 1: Let $a'_1 \in A_1$ and $b'_1 \in B_1$. If $\mathcal{P}_1 = \emptyset$ then, since $|V_1| > 2$, either $\{a'_1\} \cup A_2$ or $\{b'_1\} \cup B_2$ is a double star cutset of G. So $\mathcal{P}_1 \neq \emptyset$ and similarly $\mathcal{P}_2 \neq \emptyset$. Now suppose that $a'_1 b'_1$ is an edge. If $V_2 \setminus (A_2 \cup B_2) \neq \emptyset$ then $\{a'_1, b'_1\} \cup A_2 \cup B_2$ is a double star cutset of G. So $V_2 \setminus (A_2 \cup B_2) = \emptyset$.

Since $\mathcal{P}_2 \neq \emptyset$ there is an edge from A_2 to B_2, and hence $V_1 \setminus (A_1 \cup B_1) = \emptyset$. But this contradicts the assumption that at least one of $V_1 \setminus (A_1 \cup B_1)$ and $V_2 \setminus (A_2 \cup B_2)$ is nonempty. This completes the proof of Claim 1.

To prove (2) assume w.l.o.g. that G_1 has a double star cutset S centered at xy. By Claim 1, G_1 contains the marker path $P_2 = a_2, \ldots, b_2$. First suppose that $x, y \in V_1$. By Claim 1 no vertex of A_1 is adjacent to B_1, so S cannot contain both a_2 and b_2. W.l.o.g. S does not contain b_2. If S contains a_2 then $S \cup A_2$ is a double star cutset of G. So S does not contain a_2, and hence S is a double star cutset of G. So x or y is in P_2. By Claim 1, $\mathcal{P}_1 \neq \emptyset$ and hence S must contain a_2 or b_2. Since P_2 is of length 4 or 5, S cannot contain both a_2 and b_2. W.l.o.g. assume that S contains a_2 but not b_2. Suppose that neither x nor y coincides with a_2. Since $\mathcal{P}_1 \neq \emptyset$, vertices of A_1 are contained in different components of $G_1 \setminus S$. Then for some $u \in A_1$, $\{u\} \cup A_2$ is a double star cutset of G. Therefore, w.l.o.g. $x = a_2$. By the above argument we may also assume that S contains a vertex of A_1. But then $(S \setminus V(P_2)) \cup A_2$ is a double star cutset of G.

To prove (3) let Q be a shortest path in $\mathcal{P}_2$. By Claim 1, Q is of length at least 2. If Q is of length at least 4, then by definition of 2-join the result holds. So we may assume tha Q is of length 2 or 3. By definition of 2-join, there exists $w \in V_2 \setminus V(Q)$. If $|A_2| = |B_2| = 1$ then $V(Q)$ is a double star cutset that separates w from V_1. So we may assume that $w \in A_2$. Let $Q = x_1, \ldots, x_k$, where $x_1 \in A_2$ and $x_k \in B_2$. Let $S \in (N(x_1) \cup N(x_2)) \setminus \{w\}$. Since S cannot be a double star cutset of G, there exists a path P from w to B_2 in $G \setminus S$. By Claim 1, P contains at least 3 vertices, and hence $|V_2| \geq 6$. ∎

2-Join Decomposition Algorithm

Input: A clean graph G that has no double star cutset and no hole of length 5.

Output: Either an odd hole of G, or a family $\mathcal{L}$ of graphs that satisfies the following properties:

(1) G is odd-hole-free if and only if all graphs in $\mathcal{L}$ are odd-hole-free.

(2) No graph $F \in \mathcal{L}$ has a double star cutset or a 2-join.

(3) The number of graphs in $\mathcal{L}$ is $O(|V(G)|)$.

Step 1: Set $\mathcal{L} = \emptyset$ and $\mathcal{L}' = \{G\}$.

Step 2: If $\mathcal{L}' = \emptyset$ then stop. Otherwise, remove a graph F from $\mathcal{L}'$. If F has no 2-join, then add F to $\mathcal{L}$ and go to Step 2. Otherwise, let $V_1|V_2$ be a 2-join of F with special sets (A_1, A_2, B_1, B_2). If $V_1 \setminus (A_1 \cup B_1) = \emptyset$ and $V_2 \setminus (A_2 \cup B_2) = \emptyset$, discard F and go to Step 2. Otherwise, construct the blocks of decomposition of F, say F_1 and F_2. For $i = 1$ or 2, if $|V_i| \leq 7$, check directly whether F_i contains an odd hole. If it does, output this result and otherwise discard F_i. If $|V_i| > 7$, add F_i to $\mathcal{L}'$. Go to Step 2.

3.5 *The 2-Join Decomposition Algorithm produces the desired output, and its running time is* $O(|V(G)|^8)$.

Proof. (1) follows from 1.2 and 3.3. (2) follows by the construction of the algorithm and 3.4.

To prove that $|\mathcal{L}|$ is $O(|V(G)|)$, we construct a decomposition tree T whose root is G and whose leaves are the graphs in $\mathcal{L}$. Let F having a 2-join $V_1|V_2$ with special sets (A_1, A_2, B_1, B_2) be a nonleaf vertex of T, and let F_1, F_2 be the two blocks of the 2-join decomposition of F. Note that $V_i \neq A_i \cup B_i$ for $i = 1, 2$ by 3.4. We define $\phi(F) = |V(F)| - 12$, $\phi(F_i) = |V(F_i)| - 12$ for $i = 1, 2$. Assume first that both F_1 and F_2 appear in T. Since only F_i with $|V_i| > 7$ is added to $\mathcal{L}'$ and the marker path contains at least five vertices, $\phi(F_i) \geq 1$ for $i = 1, 2$. Furthermore, $\phi(F_1) + \phi(F_2) \leq \phi(F)$ by the fact that the marker path contains at most six vertices. Now assume that only one of the blocks F_1, F_2 belongs to T, say F_1. Then $\phi(F_1) \leq \phi(F)$ since $|V_2| \geq 6$ by 3.4 and the marker path of F_1 contains at most six vertices. Let $B_1, \ldots, B_k$ be the leaves of T. Then $k \leq \Sigma_{i=1}^{k} \phi(B_i) \leq \phi(G) = |V(G)| - 12$. This implies that the number of leaves in T is $O(|V(G)|)$.

Finding a 2-join takes time at most $O(|V(G)|^7)$ [11] and this algorithm is applied at most $O(|V(G)|)$ times, which yields an overall complexity of $O(|V(G)|^8)$. ∎

3.3 Recognition Algorithm for Odd-Hole-Free Clean Graphs

Now we present our recognition algorithm for odd-hole-free clean graphs as follows.

Input: A clean graph G such that if G contains an odd hole then G contains a spotless odd hole H^* with the property that every hole in $\mathcal{C}_G(H^*)$ has a pair of vertices at distance at least 4 in G.

Output: ODD-HOLE-FREE if G is odd-hole-free, and NOT ODD-HOLE-FREE otherwise.

Step 1: Check whether G contains a 5-hole, a 7-hole, a short odd wheel or a short 3PC. If G contains any of these then output NOT ODD-HOLE-FREE and stop.

Step 2: Apply the Double Star Decomposition Algorithm to G. If the output of this algorithm is an odd hole, then output NOT ODD-HOLE-FREE and stop. Otherwise, let $\mathcal{L}_1$ be the output family of graphs.

Step 3: Set $\mathcal{L}_2 = \emptyset$. For every graph in $\mathcal{L}_1$ apply the 2-Join Decomposition Algorithm. If the output of this algorithm is an odd hole, then output NOT ODD-HOLE-FREE and stop. Otherwise, merge the output with $\mathcal{L}_2$.

Step 4: Check if every graph of $\mathcal{L}_2$ is basic. If this is the case, output ODD-HOLE-FREE. Otherwise output NOT ODD-HOLE-FREE.

The complexity of this algorithm is $O(|V(G)|^{10})$ since it is dominated by Step 3, which takes time $O(|V(G)|^8)$ for each of the $O(|V(G)|^2)$ graphs of $\mathcal{L}_1$.

To check whether a graph G is perfect, one can check whether both G and $\bar{G}$ are odd-hole-free by the Strong Perfect Graph Theorem [8]. This can be done by applying the Cleaning Algorithm and then the Recognition Algorithm for Odd-Hole-Free Graphs, to first G and then $\bar{G}$.

Acknowledgements: We had many stimulating discussions with Michele Conforti and Giacomo Zambelli on the recognition of perfect graphs. The approach followed in this paper and the companion paper [7] grew out of our joint work with them.

References

[1] C. Berge, Farbung von Graphen deren samtliche bzw. Deren ungerade Kreise starr sind (Zusammenfassung), *Wissenschaftliche Zeitschrift, Martin Luther Universität Halle-Wittenberg, Mathematisch-Naturwissenschaftliche Reihe* (1961) 114-115.

[2] D. Bienstock, On complexity of testing for odd holes and induced odd paths, *Discrete Mathematics 90* (1991) 85-92.

[3] M. Burlet and J. Fonlupt, Polynomial algorithm to recognize a Meyniel graph, *Ann of Discrete Math 21* (1984) 225-252.

[4] V. Chvátal, J. Fonlupt, L. Sun and A. Zemirline, Recognizing dart-free perfect graphs, *SIAM Journal on Computing 31* (2002) 1315-1338.

[5] V. Chvátal and N. Sbihi, Recognizing claw-free perfect graphs, *Journal of Combinatorial Theory B 44* (1988) 154-176.

[6] M. Chudnovsky, PhD dissertation, forthcoming 2003.

[7] M. Chudnovsky, G. Cornuéjols, X. Liu, P. Seymour and K. Vušković, Cleaning for Bergeness, 2002.

[8] M. Chudnovsky, N. Robertson, P. Seymour and R. Thomas, The strong perfect graph theorem, 2002.

[9] M. Chudnovsky and P. Seymour, Recognizing Berge graphs, 2002.

[10] M. Conforti, G. Cornuéjols, A. Kapoor and K. Vušković, Balanced $0, \pm 1$ matrices II: Recognition algorithm, *Journal of Combinatorial Theory B 81* (2001) 275-306.

[11] M. Conforti, G. Cornuéjols, A. Kapoor and K. Vušković, Even-hole-free graphs, Part II: Recognition algorithm, *Journal of Graph Theory 40* (2002) 238-266.

[12] M. Conforti, G. Cornuéjols and M. R. Rao, Decompositions of balanced matrices, *Journal of Combinatorial Theory B 77* (1999) 292-406.

[13] M. Conforti, G. Cornuéjols and K. Vušković, Decomposition of odd-hole-free graphs by double star cutsets and 2-joins, preprint (2001), to appear in the special issue of *Discrete Applied Mathematics* dedicated to the Brazilian Symposium on Graphs, Algorithms and Combinatorics, Fortaleza, March 17-19, 2001.

[14] M. Conforti and M. R. Rao, Structural properties and decomposition of linear balanced matrices, *Mathematical Programming 55* (1992) 129-168.

[15] G. Cornuéjols and W.H. Cunningham, Compositions for perfect graphs, *Discrete Mathematics 55* (1985) 245-254.

[16] B. A. Reed and N. Sbihi, Recognizing bull-free perfect graphs, *Graphs Combin 11* (1995) 171-178.

[17] D. B. West, *Introduction to Graph Theory*, Prentice Hall (1996).