This paper appeared in: Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms 2002, 382 - 389

Polynomial Time Recognition of P_4-structure[*]

R. B. Hayward[†] S. Hougardy[‡] B. A. Reed[§]

Abstract

A P_4 is a set of four vertices of a graph that induces a chordless path; the P_4-structure of a graph is the set of all P_4's. Vašek Chvátal asked if there is a polynomial time algorithm to determine whether an arbitrary four-uniform hypergraph is the P_4-structure of some graph. The answer is yes; we present such an algorithm.

1 Introduction

A P_4 in a graph G is an induced path on four vertices. The P_4-*structure* of a graph G with vertex set V, denoted $P_4(G)$, is the hypergraph on V whose edges are those vertex sets which induce a P_4 in G. A hypergraph H is a P_4-*structure* if it is the P_4-structure of some graph G, and any such graph G is called a *realization* of H. Graphs with the same P_4-structure are P_4-*equal*. Since four vertices induce a path in a graph if and only if they induce a path in its complement, a graph is P_4-equal to its complement.

A hypergraph is *realizable* if it is a P_4-structure. A realizable hypergraph is *uniquely realizable* if for any two realizations each is isomorphic either to the other or to the other's complement, and *strongly uniquely realizable* if for any two realizations each is equal either to the other or to the other's complement. As the reader may verify, the P_4-structure of the graph in Figure 1 is strongly uniquely realizable, the P_4-structure of the hole with five vertices is uniquely realizable but not strongly uniquely realizable (since all 12 C_5s on a set of 5 vertices have the same P_4-structure), and the P_4-structure of the two P_4-equal graphs in Figure 2 is not uniquely realizable.

The notion of P_4-structure, introduced by Chvátal, was motivated by the study of perfect graphs. A graph is *perfect* if for every induced subgraph the chromatic number equals the clique size. (Throughout this paper, all subgraphs or subhypergraphs referred to are induced subgraphs or subhypergraphs.) Around 1960 Berge [1] introduced the notion of perfect graphs and proposed two perfect graph conjectures (PGC's) which have stimulated much research. An *odd hole* is an odd induced cycle with at least five vertices, and an *odd anti-hole* is its complement.

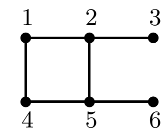

Figure 1: A realization of a strongly uniquely realizable hypergraph.

Strong PGC. *A graph is perfect if and only if it contains no odd hole or odd antihole.*

Weak PGC. *A graph is perfect if and only if its complement is perfect.*

Lovász proved the $WPGC$ in 1971 [22], while the $SPGC$ remains open. Chvátal felt that studying P_4-structure might lead to a resolution of the $SPGC$, and he made the following conjecture, later proved by Reed [24].

Semi-Strong PGC. *A graph is perfect if and only if each graph P_4-equal to it is perfect.*

Chvátal chose the name of his conjecture to reflect the fact that it is implied by the $SPGC$ (which he showed by proving that the P_4-structure of an odd hole has a unique realization), and implies the $WPGC$ (since a graph and its complement are P_4-equal).

Chvátal's motivation for introducing P_4-structure was that he hoped to use it to produce perfection certificates for perfect graphs, thereby proving that perfect graph recognition is in NP. He observed that for some classes of perfect graphs (for example, perfectly orderable graphs [6] and line graphs of bipartite graphs) it is known how to provide a certificate for a graph in the class with which the graph's perfection can be verified in polynomial time. He noted that, given the $SSPGC$, one can certify the perfection of a graph G by providing a P_4-equal graph H together with a certificate of the perfection of H. It was Chvátal's hope that this technique would be useful in certifying perfection. As we shall see, it turns out that P_4-structure is of no more use for producing perfection certificates than are

[*]Supported by NSERC Operating Grant OGP-0137764.

[†]Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada T6G 2E1, hayward@cs.ualberta.ca

[‡]Institut für Informatik, Humboldt-Universität zu Berlin, 10099 Berlin Germany, hougardy@informatik.hu-berlin.de

[§]School of Computer Science, McGill University, Montreal, Quebec, Canada H3A 2A7, breed@cs.mcgill.ca

This paper appeared in: Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms 2002, 382 - 389

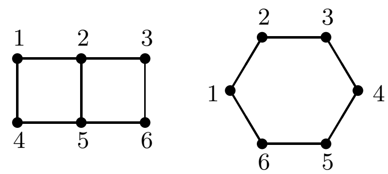

Figure 2: Two realizations of a not uniquely realizable hypergraph.

homogeneous sets. However, Chvátal's introduction of P_4-structure has generated considerable interest in how P_4's interact in perfect and minimally imperfect graphs, and motivated much work which is of interest in its own right (see for example [7, 8, 13, 15, 17, 18, 14, 19, 21]).

In this paper we establish results on P_4-structure which yield a polynomial time algorithm for recognizing P_4-structure. Because of space constraints, all proofs are omitted. Our results elucidate the relationship between P_4-structures and homogeneous sets. Before going further, we define homogeneous sets and discuss their relationship to P_4-structure realizations.

A set is *big* if it has at least two elements. A graph or hypergraph is *big* if its vertex set is big. A *homogeneous set* of a graph is a big proper vertex subset such that every vertex not in the subset sees all or none of the vertices in the subset. For example, the graphs in Figures 1 to 4 contain no homogeneous set, while the graph in Figure 6 contains the homogeneous set $\{4, 5, 6, 7\}$. A crucial lemma in Lovász's proof of the *WPGC* is that no minimal imperfect graph contains a homogeneous set. In 1974, Seinsche [25] proved that a graph G is a realization of a P_4-structure with no edges precisely if every induced subgraph of G with more than two vertices contains a homogeneous set. This led to the development of faster polynomial time algorithms for the recognition of this class of graphs (see [9]) and for determining if a graph has a homogeneous set (see [26]). In fact, more complicated techniques were eventually developed to quickly partition the vertex set of any given graph into maximal pieces which contain no homogeneous sets (see [10] [23]). Many of these results rely partially on the links between P_4-structure and homogeneous sets (see for example [26] where the notion of a P_4-tree is introduced).

It is not hard to see that the P_4-structure of a graph with a homogeneous set is not strongly uniquely realizable (cf. Observation 1.3 below), and that this complicates the problem of recognizing P_4-structure. So that we may avoid this complication, we introduce a hypergraph notion which captures the graph notion of homogeneous set. This will allow us to reduce the problem of recognizing P_4-structures to the problem of recognizing P_4-structures of graphs with no homogeneous set.

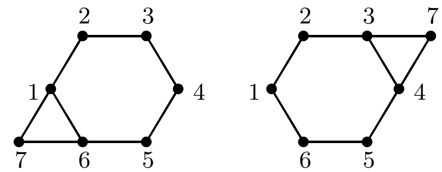

Figure 3: Illustrating degeneracy. The graph G induced by $\{1, \ldots, 6\}$ is degenerate, since it has two unequal but P_4-equal supergraphs F_1 and F_2 (shown). Notice that G has many *CSC* partitions; the *CSC* partition which gives rise to F_1 and F_2 is $C_1 = \{v_1, v_6\}$, $S = \{v_2, v_5\}$, $C_2 = \{v_3, v_4\}$.

A graph is *prime* if it has more than two vertices and no homogeneous set. Our hypergraph notion follows directly from the following observation.

Observation 1.1. *In a graph* G *with homogeneous set* S*, every* P_4 *contains 0, 1, or 4 vertices of* S*. Furthermore, if* { s,x,y,z } *induces a* P_4 *for some* s *in* S *and* x,y,z *not in* S*, then* { s′,x,y,z } *induces a* P_4 *for all* s′ *in* S. □

Thus we define an h-*set* S of a four-uniform hypergraph as a proper vertex subset such that every edge of the hypergraph has 0, 1, or 4 vertices in S, and for every two vertices s, s' in S and three vertices x, y, z not in S, $\{s, x, y, z\}$ induces an edge if and only if $\{s', x, y, z\}$ induces an edge. Now the next observation is simply a restatement of the previous.

Observation 1.2. *A homogeneous set of a graph is an* h*-set of the graph's* P_4*-structure.* □

Observation 1.3. *A* P_4*-structure with* h*-set* S *has a realization in which* S *is homogeneous, and so (the* P_4*-structure) is not strongly uniquely realizable.* □

A four-uniform hypergraph is *prime* if it has at least four vertices and no *h*-set. In light of observation 1.3 it is natural to conjecture that a realizable hypergraph is strongly uniquely realizable if and only if it is prime. This is not the case: as illustrated in Figures 2, 3, and 4 there are prime realizable hypergraphs which are not strongly uniquely realizable. However, the conjecture is not far from the truth; for example, it holds for all realizable hypergraphs which contain the P_4-structure of the graph shown in Figure 1 or the P_4-structure of an induced cycle with seven or more vertices. More generally, as Theorem 1.1 below shows, the conjecture holds as long as the hypergraph has

This paper appeared in: Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms 2002, 382 - 389

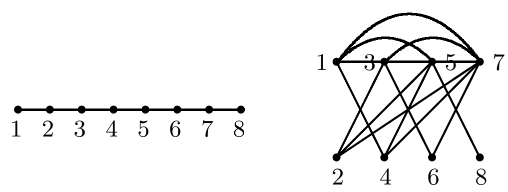

Figure 4: Two realizations of another not uniquely realizable hypergraph. Both graphs are prime. The left graph is non-degenerate; the right graph is split (and so degenerate). In the same fashion, for any k, there is a graph which is P_4-equal to P_k, so $P_4(P_k)$ is not uniquely realizable.

some induced subhypergraph with a "non-degenerate" realization. Before stating this theorem, we discuss the degenerate special case:

We say that a graph has a *CSC-partition* if its vertex set can be partitioned into cliques C_1 and C_2 and a stable set S such that there are no edges between C_1 and C_2. Complementarily, we say that a graph has an *SCS-partition* if its vertex set can be partitioned into stable sets S_1 and S_2 and a clique C such that there are all edges between S_1 and S_2. We call a graph *degenerate* if it has either a *CSC* partition or a *SCS* partition. We call such a graph degenerate because it has different but P_4-equal supergraphs, each obtained by adding one vertex as illustrated in Figure 3; such supergraphs always exist, because of the following observation.

Observation 1.4. *Let G have a* CSC *partition and let F_1 and F_2 be the two supergraphs of G obtained by adding a vertex v whose neighbourhoods are respectively C_1 and C_2. Then F_1 and F_2 are P_4-equal.*

On the other hand, we have:

Theorem 1.1. *Let* H *be a realizable hypergraph with an induced subhypergraph which has a strongly unique non-degenerate realization. Then* H *is strongly uniquely realizable if and only if* H *is prime.*

One direction of the theorem follows from Observation 1.3, so we only need prove:

Theorem 1.1′ *If a realizable prime hypergraph H has an induced subhypergraph with a strongly unique non-degenerate realization then H has a strongly unique realization.*

As discussed in §5, Theorem 1.1′ can be proved iteratively by repeatedly extending a realization of an induced subhypergraph to a realization of a larger induced subhypergraph: in each step one or two vertices are added, chosen so that the new realization is always

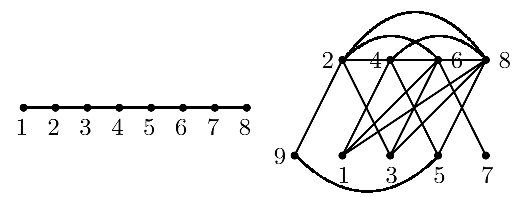

Figure 5: A non-extending prime non-degenerate subrealization of a realizable hypergraph. The graph G' on the left is a subrealization of the P_4-structure H of the graph G on the right. G' does not extend to any realization of H, because G is strongly uniquely realizable (since $G[\{v_1, \ldots, v_5, v_9\}]$ is isomorphic to the graph in Figure 1).

strongly unique. Actually, this approach allows us to prove a stronger theorem:

Theorem 1.2. *If a realizable prime hypergraph H has an induced subhypergraph with a prime non-degenerate realization G' then at most one extension of G' realizes H.*

To see that Theorem 1.2 implies Theorem 1.1′, we need only observe that strongly unique realizations cannot have homogeneous sets, and that any strongly unique subrealization G' with vertex set V' of a realizable hypergraph H has at least one extension which realizes the hypergraph (since for any realization G of H, $G[V'] = G'$ or $\overline{G'}$).

Notice that "at most one" cannot be replaced with "exactly one" in the preceding theorem, since a prime non-degenerate subrealization G' may not extend. This is illustrated in Figure 5, where G' is the graph on the left with $V(G') = \{v_1, \ldots, v_8\}$, G is the graph on the right with $V(G) = V(G') \cup \{v_9\}$, and H is the P_4-structure of G. The reader can verify that G' is prime and non-degenerate (in fact G' is the shortest path with these properties) and that G' and $G - v_9$ are P_4-equal (as noted earlier in Figure 4).

This is the skeleton of our P_4-structure recognition algorithm. We attempt to extend subrealizations which have *CSC* or *SCS* partitions in the same way. However, unique one or two vertex extensions of such graphs are not always possible (see Figure 3), so we will sometimes need to examine all the remaining vertices together, in order to decide on an appropriate extension (or set of extensions). We will need to flesh out our skeleton considerably, developing quite complicated techniques for dealing with graphs that have *CSC* or *SCS* partitions. Particularly troublesome in this regard are the *split* graphs, namely those graphs which have a partition into a clique and a stable set (and hence both a *CSC* and *SCS* partition).

This paper appeared in: Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms 2002, 382 - 389

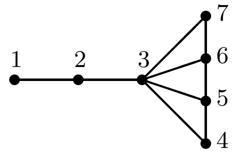

Figure 6: A graph with a homogeneous set.

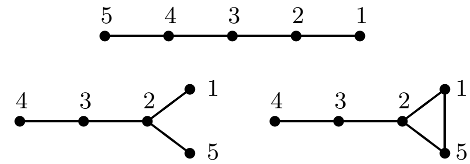

Figure 7: Three P_4-equal graphs. Only the top is prime.

We close this introductory section by remarking that Hayward [14] has carried out a similar study of the P_3-structure of graphs, and that remarks on the F-structure of graphs for various F can be found there as well as in [4], [5] and [11].

2 Dealing with *h*-sets

Our observations above show that in constructing realizations via one or two vertex extensions, things are much easier if the hypergraph we are trying to realize is prime. The following lemmas show that we can restrict ourselves to this case, by busting up G into a set of prime subhypergraphs and combining their realizations.

LEMMA 2.1. *There is an algorithm which determines whether an input hypergraph is prime, and returns an* h-*set if it is not.*

By *substituting* a graph F for a vertex v in a graph G, we mean creating the graph G' with vertex set $V(G) - v \cup V(F)$ whose edges consist of all edges of $G - v$, all edges of F, and all edges (wf) such that f is in F and w is adjacent to v in G.

LEMMA 2.2. *A four-uniform hypergraph H with vertex set V and* h-*set S is realizable if and only if the hypergraphs induced by $V-(S-x)$ and S are realizable. Furthermore, substituting any realization of $H[S]$ for vertex x in any realization of $H[V-(S-x)]$ yields a realization of H.*

In performing our extensions, we will want to know how a one vertex extension of G can not be prime if G is prime. The following easy observation is important.

OBSERVATION 2.1. *Let $G = (V, E)$ be a realization of a prime hypergraph H which has a non-prime superhypergraph $H + v$. Then there is a supergraph $G + v$ of G realizing $H + v$ in which either (i) v is a twin of some vertex in V or (ii) v is adjacent to all or no vertices of V.*

The observation follows from the fact that the only possible h-set of $H + v$ is V or $\{v, x\}$ for some x in V, as any other h-set yields an h-set of H. Observe that an h-set of a hypergraph need not be a homogeneous set in every realization of the hypergraph. See for example Figure 7. This is an important observation to consider in designing an algorithm to construct all realizations of a given P_4-structure.

Finally, the following result will be important in analyzing the second possibility discussed in this observation:

LEMMA 2.3. *Let* H *be a realizable hypergraph with vertex set* V*, let* v *be a vertex of* V *in no edge of* H*, let* F *be a realization of* H*, and let* G *be the subgraph of* F *induced by* V−v*. If* G *is prime then either:*

(a) in F*,* v *is* V−v*-universal, or*

(b) in F*,* v *is* V−v*-null, or*

(c) in F*,* v *sees a clique* C *and misses a stable set* S*, and so* F *(and also* G*) is a split graph.*

3 The Key Lemma

As mentioned earlier, the essential step in our iterative P_4-structure recognition algorithm involves extending a subrealization G' of a candidate hypergraph H by one or two vertices chosen so as to ensure that any such extension is unique. We need to insist that all these subrealizations are prime so that we can apply Lemma 2.3 as well as some more powerful tools developed in this section. In §2 we took our first step in this direction when we showed that we can insist that H is prime. In §2 we also discussed under what conditions we can extend a realization and maintain primality. In this section we determine under what circumstances we can extend a subrealization by one vertex and still maintain primality and uniqueness. As we shall see we need to deal with the exceptional cases pointed out in the previous section (homogeneous sets of size 2 and vertices in no hyperedges) and those pointed out in the first section (CSC and SCS partitions). In the next section we consider the corresponding two vertex extensions.

For a realizable hypergraph H with a realization G' of $H[V']$, we call a vertex x of $V - V'$

- an *obstructor* if no extension of G' realizes $H[V'+x]$

This paper appeared in: Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms 2002, 382 - 389

- a *unique extender* if exactly one extension of G' realizes $H[V'+x]$
- an *extremist* if x is in no edge of $H[V'+x]$
- a *clone* of some x' in V' (and x' is a *mate* of x) if $\{x,x'\}$ is an h-set of $H[V'+x]$
- a *CSC-swapper* if G' has a *CSC* partition such that for $j=1,2$ the extensions G_j in which $N(x)=C_j$ realize $H[V'+x]$
- an *SCS-swapper* if G' has an *SCS* partition such that for $j=1,2$ the extensions G_j in which $N(x)=C\cup S_j$ realize $H[V'+x]$.

We say that x is a *swapper* if it is a *CSC* swapper or an *SCS* swapper. This term is of course motivated by Observation 1.4: in extending G', the neighbourhood of x can 'swap' between C_1 and C_2 (in the case of a *CSC* partition) or $C\cup S_1$ and $C\cup S_2$ (in the case of an *SCS* partition).

We can now describe all possible one vertex extensions of a realizable hypergraph.

LEMMA 3.1. **(Key Lemma)** *Let $G'=(V',E')$ be a prime subrealization of a four-uniform hypergraph $H=(V,Q)$ with $V=V'+x$. Then exactly one of the following holds:*

(a). x *is an obstructor*

(b). x *is a unique extender*

(c). x *is a extremist and G' has the two extensions in which x is V'-extreme and has at least three extensions only if G' is split (by Lemma 2.3)*

(d). x *is a clone and not a swapper in which case* x *is a clone of exactly one vertex, say x', and G' has only the two extensions in which x and x' are twins*

(e). x *is a* CSC *swapper and not an extremist, in which case G' has 2,3,or 4 extensions,*

(f). x *is a* SCS *swapper and not an extremist, in which case G' has 2,3,or 4 extensions,*

Remark: if G' has no *CSC* partition or *SCS* partition then exactly one of (a) through (d) occurs, since, by Lemma 2.3, (c) and (d) cannot both occur (as this would imply that G' is split and so has a *CSC* partition). Also, if G' has a *CSC* or *SCS* partition but is not split, then the statement of the lemma can be simplified somewhat.

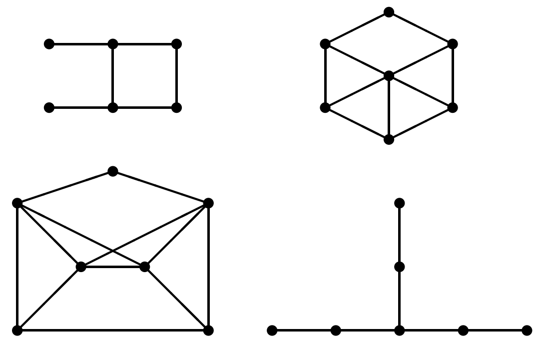

Figure 8: Graphs with strongly unique non-degenerate P_4-structures.

4 Two-vertex Extensions

Clearly, Theorem 1.2 can easily be proved by recursively applying the following result, since every extension of a non-degenerate realization is non-degenerate.

THEOREM 4.1. *If a realizable prime hypergraph H with vertex set V has an induced subhypergraph $H[V']$ with a prime non-degenerate realization G' then there is a set R of at most two vertices of $V-V'$ such that $H[V'+R]$ has at most one realization extending G' and furthermore any such realization is prime.*

For example, each graph in Figure 8 is a strongly unique non-degenerate realization of its P_4-structure, so Theorem 1.1 implies that any prime graph with one of these graphs as an induced subgraph is a strongly unique realization of its P_4-structure.

We actually provide an algorithmic proof of this theorem which allows us to determine for any prime non-degenerate subrealization G of a prime hypergraph H whether there is realization of H extending G. This algorithm relies on a reasonably straightforward polynomial-time procedure which generates all extensions of G to $V(G)+v$ for any vertex v (the details of which we omit), and a lemma which guarantees that for any prime non-degenerate extendible subrealization of a prime realizable hypergraph, there is always an extending vertex set of size one or two, such that the extension is prime and unique.

The key to this lemma is Lemma 3.1 which tells us that since there are no swappers, we can determine that either the subrealization G is not extendible ((a) holds for some x), or for some x there is a unique realization of $G+x$ ((b) holds), or for every vertex x, x is a clone or an extremist. We can assume that this last case holds, as otherwise we can perform a one vertex extension. Furthermore, since G is not split every extremist sees all

This paper appeared in: Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms 2002, 382 - 389

or none of $V(G)$ in every extension. Thus, there must exist some clone as otherwise $V(G)$ is a homogeneous set in every realization of H and hence an h-set of H, a contradiction.

Now, suppose that x is a clone of some x' and let $S(x')$ be the set of clones of x'. If in every realization, every vertex of $H - S(x')$ is adjacent to either all or none of $S(x')$ then $S(x')$ is an h-set of H, a contradiction. So, we can find a vertex v not in $S(x')$ and a vertex x^* in S' such that in some realization of H, v is adjacent to exactly one of x and x^*. It is straightforward to verify that v and x^* are the desired pair of vertices whose addition yields a larger subhypergraph with a unique prime realization which is an extension of G (we omit further details).

5 The Degenerate Case

If we are trying to extend a prime subrealization G which is degenerate but not split, then we can proceed as above, provided that there are no swappers for G. In fact, similar reasoning allows us to show that we can always find a unique two vertex extension of a prime non-split subrealization to a new prime subrealization unless every vertex is a swapper.

Dealing with the case in which every vertex is a swapper is complicated. It is particularly tricky when G has both an SCS and CSC partition and hence, possibly, swappers of both types. The following easy observation simplifies our analysis of this case:

Observation 5.1. *G has both an SCS and a CSC partition if and only if for some vertex v, $G - v$ is split.*

By this observation, provided a degenerate G is not obtained from a split graph by adding a vertex, then we can assume (by passing to the complement) that G admits an CSC partition but no SCS partition. The techniques developed to treat this case include two new decompositions and some ad-hoc arguments. We present one of the decompositions in §7; In the next section we motivate the decomposition and illustrate the ad-hoc arguments by discussing the case when G is a C_6.

We consider only briefly the case when $G - v$ is split for some v. We remark that the authors [unpublished] and independently Brandstädt and V.B. Le [3] have obtained a polynomial time algorithm to determine if G has a split representation. Our techniques for dealing with nearly split G use techniques developed to search for split representations. They also use the fact that if we can find a small (namely constant size) prime subhypergraph which permits no nearly-split realizations (for example $P_4(C_6)$), then by trying all of the finitely many possible realizations of this hypergraph, we bypass the nearly split case.

6 Extending a C_6 realization

In this section, we examine how to proceed if G is C_6 and every vertex of $H - G$ is a swapper. As we will point out, we use many ideas which apply to extending any non nearly-split subrealization G for which every vertex of $G - H$ is a swapper.

So, let G be a subrealization of some P_4-structure H which induces a C_6 with edge set $\{01, 12, 23, 34, 45, 50\}$ and such that every vertex of $H - G$ is a swapper. Since the only CSC partitions of a C_6 consist of two edges and a stable set containing two antipodal vertices, it follows that $V(H) - V(G)$ can be partitioned into sets S_0, S_1, S_2 where for any realization of H and vertex x in S_i we have that $N(x) \cap G$ is either $i, i+1$ or $i+3, i+4$ (where addition is modulo 6).

Note that if $x \in S_i$ and $y \in S_j$ then there are at most 8 candidate choices for an extension of G to a realization of the subhypergraph of H induced by $V(G) + x + y$, for we need only specify whether xy is an edge and then specify the choices of $N(x) \cap G$, and $N(y) \cap G$. Furthermore, if $i \neq j$ then none of these candidates are degenerate as the reader may easily verify. So, in this case, we can use the algorithm of §4 to compute all the extensions of each of these candidates to a realization of H and thereby determine if H has any realization extending G. Thus, we can assume without loss of generality that S_0 and S_1 are empty (In the same way, for general non nearly-split G, our analysis can often be simplified by noting that for many choices of 2-vertex extensions we will obtain a non-degenerate supergraph of G, for which we can easily test extendability)

Now, for any realization F of H we let $S_1'(F)$ be the set of vertices of $S_1 = V(H) - V(G)$ adjacent to $1, 2$ in F and let $S_1^*(F)$ be the subset of S_1 adjacent to $4, 5$. We note that xy is an edge between S_1' and S_1^* precisely if $1, x, y, 4$ is an edge of H. More strongly, xy is an edge between $S_1'(F')$ and $S_1^*(F')$ for every realization F' of H extending G precisely if $1, x, y, 4$ is an edge of H. Thus, we can determine the set of such *cross edges* in polynomial time just by examining H.

If there are no such cross edges, then for each vertex w of $S'(F)$, $N(w) - S_1'(F)$ is exactly $\{1, 2\}$. So, $S_1'(F)$ must contain only one vertex or it is a homogeneous set. Similarly $S_1^*(F)$ consists of only one vertex. Thus H has exactly two more vertices than G, and as discussed above there are only 8 candidate extensions of G to a realization of H, so we can test the extendability of G easily. Thus, we can assume that there are some cross edges (indeed for general non nearly-split G, if there are no cross edges then we can easily determine if G is extendible to a realization of H as the existence of many possible extensions implies that there is a homogeneous set).

If xy is a cross-edge then there are exactly two realizations of the hypergraph induced by $V(G) \cup \{x, y\}$ extending G. In one of these $N(x) = \{1, 2, y\}$ and $N(y) = \{4, 5, x\}$. The other is obtained from this labelled graph by swapping label 1 with 5 and label 2 with 4. In the same way, if U is a component of the (bipartite) graph induced by the cross edges, then there are either zero or two realizations of H extending G and if there are two realizations then one is obtained from the other by swapping label 1 with 5 and label 2 with 4. Furthermore, we can find the two candidate extensions of G to the subhypergraph induced by $V(G) \cup U$ in polynomial time.

In the same vein, if the cross-edge graph has a fixed number of components then there are a fixed number of ways of extending G to a realization of the hypergraph induced by $V(G)$ together with all the endpoints of the cross edges, and we can generate all these candidates in polynomial time. As remarked earlier, the case in which there are no cross edges is easier so we are not recursing here but solving a bounded number of easier problems which we already know can be resolved in polynomial time.

Our only difficulty then, is when there are many components in the cross edge graph. In this case, we have to decompose our problem into subproblems. In the next section, we consider the case in which every component of the cross edge graph is a single edge, and we know that the graph induced by the vertices in this cross edge matching consists of a clique, a stable set, and a matching between them which is precisely the set of cross edges. This may seem like a very special case, but is in fact one of only two that we have to treat when considering the cross edges of any non nearly-split G.

7 A New Decomposition

A spiked clique is a graph which consists of a clique C with k vertices $c_1, .., c_k$, a stable set S with k vertices $s_1, ..., s_k$ such that the edges between the clique and the stable set are precisely the matching $s_1c_1, ..., s_kc_k$.

We remark that a spiked clique is P_4-equal to any graph obtained by swapping the labels on any edge of the matching or on any set of edges of the matching. As shown by Hougardy [21], a spiked clique of size at least three is P_4-equal exactly to these graphs and their complements.

We call the P_4-structure of a spiked clique *homogeneous* if the following properties hold:

(i) For each edge e of H intersecting $C \cup S$ in exactly one vertex x, $e - x + y$ is an edge of H for all y in $C \cup S$.

(ii) For each edge e of H intersecting $C \cup S$ in exactly two vertices, $e \cap (S \cup C)$ is $\{s_i, c_i\}$ for some i, and $e - s_i - c_i + s_j + c_j$ is an edge of H for every j with $1 \leq j \leq k$.

(iii) For each edge e of H intersecting $C \cup S$ in a set W of exactly three vertices, W contains c_i and s_i for some i; furthermore, for any j and for every set T of three vertices in $S \cup C$ containing both s_j and c_j, $e - W + T$ is an edge of H.

It turns out that if there is a homogeneous spiked clique P_4-structure with at least six vertices and with partition C, S in H, then in any realization F of H, C is a homogeneous set of $F - S$ and S is a homogeneous set of $F - C$. Furthermore, given any realization, swapping the label of c_i with that of s_i gives another realization.

These observation allow us to replace the P_4-structure of any homogeneous spiked clique of size at least eight by a substructure corresponding to a spiked clique of size six. Having solved this problem we simply replace the original larger spiked clique in such a way that S is homogenous in $F - C$ and C is homogeneous in $F - S$ to obtain a realization of all of H.

8 Conclusions

We have described a polytime algorithm which recognizes P_4-structure, answering an open question of Chvátal. It follows that for any graph F with at most four vertices, recognizing F-structure can be solved in polytime. It would be interesting to know whether this is also the case for all larger graphs, or whether there is some graph F^* such that recognizing F^*-structure is NP-hard.

Acknowledgements

The first author thanks Mike Hallett and Paul Kearney for comments on earlier versions of this work.

References

[1] C. Berge, Färbung von Graphen deren sämtliche bzw. deren ungerade Kreise starr sind, *Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg Math.-Natur. Reihe* (1961), 114–115.

[2] C. Berge and V. Chvátal (Eds.), “Topics on Perfect Graphs”, *Annals of Discrete Math.* **21**, North-Holland, Amsterdam 1984.

[3] A. Brandtstädt and V. B. Le, Split-perfect graphs: characterizations and algorithmic use, to appear in *Disc. Appl. Math.*

[4] A. Brandtstädt, V. B. Le, S. Olariu, Recognizing the P_4-structure of block-graphs. *Technical Report Universität Rostock*, 1996.

[5] A. Brandtstädt, V. B. Le, S. Olariu, Linear-time recognition of the P_4-structure of trees. *Rutcors Research Report* 19-96 (1996).

[6] V. Chvátal, Perfectly ordered graphs, in [2].

[7] V. Chvátal, C. T. Hoàng, On the P_4-structure of perfect graphs I. Even decompositions, *J. Combin. Theory Ser.* **B** 39 (1985), 209–219.

[8] V. Chvátal, On the P_4-structure of perfect graphs III. Partner decompositons, *J. Combin. Theory Ser.* **B** 43 (1987), 349–353.

[9] D. G. Corneil, Y. Perl, L. Stewart, Cographs: recognition, application and algorithms, *Congr. Num.* 43 (1984), 249–258.

[10] A. Cournier, M. Habib, A new linear algorithm for modular decomposition, *LNCS* 787 (1994), 68–84.

[11] G. Ding, Recognizing the P_4-structure of a tree, *Graphs Combin.* **10** (1994), 323–328.

[12] M. Golumbic, "Algorithmic graph theory and perfect graphs", Academic Press, San Diego, 1980.

[13] R. B. Hayward, On the P_4-structure of perfect graphs. III. Weakly triangulated graphs, *McGill University SOCS, Technical Report 83.20*, Montreal, Canada, 1983.

[14] R. B. Hayward, Recognizing P_3-structure: A switching approach, *J. Combin. Theory B* **66** (1996), 247–262.

[15] R. B. Hayward, W. J. Lenhart, On the P_4-structure of perfect graphs. IV. Partner graphs, *J. Combin. Theory B* 48 (1990), 135–139.

[16] R. B. Hayward, W. J. Lenhart, Bichromatic P_4 Composition Schemes for Perfectly Orderable Graphs, manuscript.

[17] C. T. Hoàng, On the P_4-structure of perfect graphs II. Odd decompositions, *J. Combin. Theory B* 39 (1985), 220–232.

[18] C. T. Hoàng, On the sibling-structure of perfect graphs, *J. Combin. Theory B* 49 (1990), 282–286.

[19] C. T. Hoàng, S. Hougardy, F. Maffray On the P_4-structure of perfect graphs V. Overlap graphs, *J. Combin. Theory B* 67 (1996), 212–237.

[20] C. T. Hoàng & N. Khouzam, On brittle graphs, *J. Graph Theory* **12** (1988) 391-404.

[21] S. Hougardy, On the P_4-Structure of Perfect Graphs, *Shaker, Aachen, Germany* 1996

[22] L. Lovász, Normal hypergraphs and the Perfect Graph Conjecture, *Discrete Math.* **2** (1972), 253–267.

[23] R. M. McConnell, J. P. Spinrad, Linear time modular decomposition of undirected graphs and efficient transitive orientation of comparability graphs, in *Proceedings of the Symposium on Discrete Mathematics* (1993), 536–545.

[24] B. A. Reed, A semi-strong perfect graph theorem, *J. Combin. Theory B* **43** (1987) 223-240.

[25] D. Seinsche, On a property of the class of n–colorable graphs, *J. Comb. Theory B* **16** (1974), 191–193

[26] J. Spinrad, P_4-trees and substitution decomposition, *Discrete Appl. Math* **39** (1992), 265–291.