On Realizations of Point Determining Graphs, and Obstructions to Full Homomorphisms

Tomás Feder
268 Waverley St., Palo Alto, CA 94301, USA
tomas@theory.stanford.edu,
and
Pavol Hell
School of Computing Science
Simon Fraser University
Burnaby, B.C., Canada V5A 1S6
pavol@cs.sfu.ca

Abstract

A graph is point determining if distinct vertices have distinct neighbourhoods. A realization of a point determining graph H is a point determining graph G such that each vertex-removed subgraph $G - x$ which is point determining, is isomorphic to H. We study the fine structure of point determining graphs, and conclude that every point determining graph has at most two realizations.

A full homomorphism of a graph G to a graph H is a vertex mapping f such that for distinct vertices u and v of G, we have uv an edge of G if and only if $f(u)f(v)$ is an edge of H. For a fixed graph H, a full-H-colouring of G is a full homomorphism of G to H. A minimal H-obstruction is a graph G which does not admit a full H-colouring, such that each proper induced subgraph of G admits a full H-colouring. We analyze minimal H-obstructions using our results on point determining graphs. We connect the two problems by proving that if H has k vertices, then a graph with $k + 1$ vertices is a minimal H-obstruction if and only if it is a realization of H. We conclude that every minimal H-obstruction has at most $k + 1$ vertices, and there are at most two minimal H-obstructions with $k + 1$ vertices.

We also consider full homomorphisms to graphs H in which loops are allowed. If H has ℓ loops and k vertices without loops, then every minimal H-obstruction has at most $(k + 1)(\ell + 1)$ vertices, and, when

both k and ℓ are positive, there is at most one minimal H-obstruction with $(k+1)(\ell+1)$ vertices.

In particular, this yields a finite forbidden subgraph characterization of full H-colourability, for any graph H with loops allowed.

1 Introduction

This paper can be read in two different ways. For the readers interested in point determining graphs, we introduce tools and machinery that allows us to analyze the fine structure of point determining graphs, and their realizations. The surprizing result we obtain at the end of Section 2, following a sequence of lemmas, is that a graph can have at most two realizations. Along the way, we obtain other results previously observed about point determining graphs [12].

For the readers interested in full H-colourings (or M-partitions), we prove in Section 3, that full H-colourability (the existence of a full H-colouring) has a finite forbidden subgraph characterization, for any graph H, including graphs with loops allowed. Refining this result, we analyze the minimal forbidden subgraphs, called minimal H-obstructions. It turns out that we can obtain good bounds on the size of these minimal obstructions; the best of these are obtained by applying the above result on realizations. However, the proof is written in such a way that the reader not interested in the best possible bounds on the size of minimal H-obstructions can read and understand those parts of the proofs in Section 3 which yield the easiest bounds, without having read the detailed analysis of point determining graphs, or without reading the more technical arguments.

We first focus on graphs *without loops.* Vertices u and v of a graph G are *similar* in G, if they have exactly the same set of neighbours, other than u and v, in G. Note that similar vertices may be adjacent or nonadjacent. It is easy to see that similarity is an equivalence relation on the vertices of G, and that each equivalence class induces an independent set or a clique. If vertices u, v are not similar, there exists a vertex $w \neq u, v$ adjacent to exactly one of them; we sometimes say that this vertex w *distinguishes* x and y.

A graph without two nonadjacent similar vertices is called *point determining.* Thus a graph is point determining if and only if distinct vertices have distinct neighbourhoods. (Since the neighbourhood of a vertex of a graph without loops does not include the vertex itself, adjacent vertices automatically have distinct neighbourhoods.) Point determining graphs were investigated by D. Sumner and others, in the seventies [2, 8, 12, 13, 14].

Sumner defined the *nucleus* of a point determining graph G to be the subgraph induced by all vertices v such that $G-v$ is also point determining; he proved that the nucleus of a connected graph has at least two vertices [12]. This will also follow from our results, cf. Corollary 2.3.

Let G and H be point determining graphs. We say that G is a *realization* of H if, for all vertices x in the nucleus of G, the vertex-deleted subgraph $G-x$ is isomorphic to H. When we speak of G being a realization of H, it will always be assumed that G and H are point determining (and we will not repeat this assumption.)

Our main result states that every point determining graph H has at most two realizations. It is an interesting open question to classify which point determining graphs have zero, one, or two realizations. (All three cases can occur.)

Let G and H be any graphs. A *homomorphism* of G to H is a mapping f of the vertices of G to the vertices of H such that $f(u)f(v)$ is an edge of H if uv is an edge of G. A homomorphism f is *full* if for distinct vertices u and v we have $f(u)f(v)$ an edge of H if only if uv is an edge of G.

We will now expand our definition of homomorphisms and full homomorphisms to allow the graph H (but not the graph G) to have *loops*. If a homomorphism f of G to H has $f(u)=f(v)$ for adjacent vertices u,v of G, then $f(u)$ must have a loop of H. If w has a loop in H, then the set of vertices u of G with $f(u)=w$ must form a clique of G. When we use the word *graph* without specifying that loops are allowed, the graph is assumed to be loopless. In particular, in Section 2, all graphs are loopless.

A homomorphism of G to H is also called an *H-colouring* of G. (Note that a K_n-colouring is exactly an n-colouring in the usual sense.) A full homomorphism of G to H is also called a *full H-colouring* of G.

Suppose H is a fixed graph with loops allowed. The decision problem "given an input graph G, does G admit an H-colouring" is polynomial time solvable if H is bipartite or has a loop, and is NP-complete otherwise [10, 11]. The decision problem "given an input graph G, does G admit a full H-colouring" is polynomial time solvable for every H; this follows from [5], via a translation of the full H-colouring problem to a certain matrix partition problem, as explained in the next paragraph. We shall show, more generally, that for every graph H, including graphs with loops allowed, there exists a finite forbidden subgraph characterization of graphs G which admit a full H-colouring. Specifically, a *minimal H-obstruction* is a graph G which does not admit a full H-colouring, such that each proper induced subgraph of G admits a full H-colouring. (A graph G which does not admit a full homomorphism to H will sometimes be simply called an *H-obstruction.*)

Applying our results on point determining graphs, we shall show that each minimal H-obstruction has at most $(k+1)(\ell+1)$ vertices, there are at most two minimal H-obstructions with $(k+1)(\ell+1)$ vertices, and when $k \geq 1$ and $\ell \geq 1$ there is at most one minimal H-obstruction with $(k+1)(\ell+1)$ vertices which we explicitly describe. A weaker version of these results is proved directly without relying on the analysis of realizations of point determining graphs, and those parts of Section 3 can be read independently of Section 2. The fact that minimal H-obstructions are always bounded in size has independently been proved, in the more general context of constraint satisfaction problems, in [1], and, in a particularly simple direct way, in [15], cf. also [7, 9].

In [3, 4, 5, 6], we have studied matrix partition problems of the following type, cf. [9, 11]. Let M be a symmetric m by m matrix over $\{0, 1, *\}$. An *M-partition* of a graph G is a partition of the vertices of G into m *parts*, indexed by the rows (and columns) of the matrix M, such that for distinct vertices x and y of the graph G, placed in parts i and j (possibly with $i = j$) respectively, we have that xy is an edge of G when $M(i, j) = 1$, and xy is not an edge of G when $M(i, j) = 0$. (If $M(i, j) = *$, then xy may or may not be an edge in G.) If M is the adjacency matrix of a graph H with loops allowed, then an M-partition of G is precisely an H-colouring of G. Moreover, if M has no $*$'s, then it always is the adjacency matrix of a graph H with loops allowed. Thus the results of this paper give a forbidden subgraph characterization of the M-partition problem for matrices M without $*$'s. Forbidden subgraph characterizations are known to exist for certain M-partition problems restricted to perfect graphs [3]. Our results in this paper are the first forbidden subgraph characterization of a large class of M-partition problems for unrestricted graphs. For more recent results on forbidden induced subgraph characterizations of matrix partition problems, see [7].

2 The Structure of Point Determining Graphs

Let G be a point determining graph. A *triple* T of G consists of a vertex x of G, called the *red vertex* of T, and an unordered pair $\{y, z\}$ of vertices of G, called the *green vertices* of T, such that $G - x$ is not point determining, and y and z are nonadjacent similar vertices of $G - x$. We shall write $T = (x, \{y, z\})$ to denote the triple. Note that x is the only vertex of G which distinguishes vertices y and z.

Lemma 2.1 *Let G be a point determining graph, and let T_1 and T_2 be two triples of G. Then one of the following cases must occur:*

1. *T_1 and T_2 share no vertices;*
2. *T_1 and T_2 share a single vertex x, which is red in both triples or green in both triples;*
3. *T_1 and T_2 share exactly two vertices x, y, with x red in T_1 and y red in T_2;*
4. *T_1 and T_2 share all three vertices x, y, z, with x red in T_1 and y red in T_2.*

Proof. Consider two triples that share a vertex z which is red in one triple and green in the other, say triples $T_1 = (z, \{u, v\})$ and $T_2 = (x, \{y, z\})$. If $\{x, y\}$ and $\{u, v\}$ are disjoint, then since z is is the unique vertex distinguishing u, v, the vertex y must be adjacent to both or to neither of u, v. This means one of u, v distinguishes y and z, which contradicts the fact that $(x, \{y, z\})$ is a triple of G (i.e., x is the only vertex of G distinguishing y and z). If y but not x is one of u, v, say, $y = u$ and $v \neq x$, then v is not adjacent to $u = y$, so v is not adjacent to z, since $(x, \{y, z\})$ is a triple and $v \neq x$. The vertices $u = y$ and z are not adjacent either, as $(x, \{y, z\})$ is a triple; this contradicts the fact that $(z, \{u, v\})$ is a triple. Therefore x must be one of u, v and we obtain the last two cases of the lemma.

Consider next two triples that share a vertex that is red in both triples. If they share at least one other vertex, say triples $(x, \{y, z\})$ and $(x, \{z, t\})$, with $t \neq y$. Then we must have y, z, t pairwise nonadjacent and have the same set of neighbours other than x. Moreover, x is either adjacent to both y and t or to neither y nor t, so y and t are similar, and G is not point determining. Thus we must have the second case of the lemma.

Finally, consider two triples that share a vertex that is green in both triples. They cannot share another green vertex, as triples $(x, \{y, z\})$ and $(t, \{y, z\})$ with $t \neq x$, since this contradicts the definition of a triple. The cases where they share another vertex that is red in one triple and green in the other, or red in both, have been considered above. Thus they must share only the green vertex, and we obtain the second case of the lemma.

Otherwise, the two triples do not share any vertices and we have the first case of the lemma. □

Let G be a point determining graph. We say that triples of G are in relation R if they share at least two vertices. Let R^* be the transitive closure of R. Each equivalence class C of the relation R^* will be called a *triple component* of G. Note that C is a set of triples of G; we shall also view C as a subgraph of G consisting of all the vertices of the triples in C and all the edges of G they induce. A vertex of C will be classified as *green*, if it is green in all the triples of C, *red* if it is red in all the triples of C, *red-green* if it is red in some triples of C and green in some triples of C, and *blue* if it does not lie in any triple of C. The *attachment points* of a triple component C are the vertices of C that belong also to triples of G not in C.

We conclude from Lemma 2.1 that a red-green vertex of a triple component C cannot be an attachment point since this would mean that some two triples share a single vertex that is red in one and green in the other. In fact, we can derive from Lemma 2.1 quite a lot of structural information about triple components of a point determining graph G.

A *type-one* triple component of G consists of vertices $u_1, \dots, u_r, v_1, \dots, v_r$, with $r \geq 2$, and triples $(u_{i+1}, \{v_i, v_{i+1}\})$ and $(v_i, \{u_i, u_{i+1}\})$, for all $i = 1, \dots, r-1$. In a type-one triple component C all vertices other than u_1 and v_r are red-green. Thus only u_1 or v_r can be attachment points; note that both are green. The edges of C must be either all u_iv_j with $j < i$ or all all u_iv_j with $j \geq i$. Indeed, if u_1v_1 is an edge of C, then u_2v_1 is not an edge since $(v_1, \{u_1, u_2\})$ is a triple of C; then u_1v_2 must also be an edge since $(u_2, \{v_1, v_2\})$ is a triple of C, and so on, yielding the edges u_iv_j with $j \geq i$ and no others. On the other hand, if u_1v_1 is a nonedge of C, then u_2v_1 must be an edge, u_1v_2 a nonedge, and so on, yielding the edges u_iv_j with $j < i$ and no others.

A *type-two* triple component of G consists of vertices $u_1, \dots, u_r, u_{r+1}$, $v_1, \dots, v_r$, with $r \geq 1$, and triples $(u_{i+1}, \{v_i, v_{i+1}\})$ for all $i = 1, \dots, r-1$, and $(v_i, \{u_i, u_{i+1}\})$ for all $i = 1, \dots, r$. In a type-two triple component C with $r \geq 2$ all vertices other than u_1 and u_{r+1} are red-green. Thus only u_1 or u_{r+1} can be attachment points; note that again both are green. When $r = 1$, all three vertices u_1, u_2, v_1 can be attachment points, and u_1, u_2 are both green, v_1 is red. In either case, we may assume that the edges of C consist of all u_iv_j with $j < i$ and no others. (In the symmetric case where the edges are all u_iv_j with $j < i$, we can simply invert the order of the u_i's and of the v_j's.)

A *type-three* triple component of G consists of vertices $u_1, \dots, u_r, u_{r+1}$, $v_1, \dots, v_r$, with $r \geq 1$, and triples $(u_{i+1}, \{v_i, v_{i+1}\})$ for all $i = 1, \dots, r-1$, and $(v_i, \{u_i, u_{i+1}\})$ for all $i = 1, \dots, r$, plus one additional triple $(u_{r+1}, \{v_r, u_1\})$. In a type-three triple component C with $r \geq 2$, all vertices other than u_1 are red-green; thus only u_1 can be an attachment point, and it is green. When $r = 1$, all three vertices u_1, u_2, v_1 can be attachment points, and v_1, u_2 are both red, u_1 is green. In this case, the edges of C must be all u_iv_j with $j < i$.

Note that only type-two and type-three triple components can have $r = 1$, i.e., consist of just three vertices. As detailed above, they are different from other triple components in that they have red vertices (rather than red-green vertices). They will play an important role in the proof below. We shall call a triple component with $r = 1$ *small*.

Note that if C is a type-one, type-two, or type-three triple component, the vertices u_i have the same neighbours outside C, and the vertices v_j have the same neighbours outside C. For type-three triple components C, we see that in fact all vertices of C must have the same neighbours outside C.

Theorem 2.2 *Every triple component of a point determining graph must be a type-one, type-two, or type-three triple component.*

Proof. Let C be a triple component of a point determining graph G. If C has a green vertex, denote it u_1, and choose a triple $(v_1, \{u_1, u_2\})$ of C containing u_1. If u_2 is also green in C, then C is a small type-two triple component, by Lemma 2.1. In general, assume we have obtained vertices $u_1, \ldots, u_r, u_{r+1}$, and $v_1, \ldots, v_r$, and triples $(u_{i+1}, \{v_i, v_{i+1}\})$ for all $i = 1, \ldots, r-1$, and $(v_i, \{u_i, u_{i+1}\})$ for all $i = 1, \ldots, r$. If u_{r+1} is a green vertex of C, then C is a type-two triple component of G by Lemma 2.1. If u_{r+1} is red in some triple, then Lemma 2.1 implies that this triple is $(u_{r+1}, \{v_r, z\})$ for some z. If $z = u_1$ then we have a type-three triple component. Otherwise z must be a vertex not yet enumerated, since all other enumerated vertices are red in some triple having no other common vertex with the triple $(u_{r+1}, \{v_r, z\})$. Thus we let $v_{r+1} = z$. In general, assume we have obtained $u_1, \ldots, u_r$, and $v_1, \ldots, v_r$, and triples $(u_{i+1}, \{v_i, v_{i+1}\})$ and $(v_i, \{u_i, u_{i+1}\})$ for all $i = 1, \ldots, r-1$. If v_r is a green vertex of C, then C is a type-one triple component by Lemma 2.1. If v_r is red in some triple, then Lemma 2.1 implies that this triple is $(v_r, \{u_r, z\})$ for some z not yet enumerated. (In this case $z = u_1$ is not possible, as the neighbours of v_r in C include both or neither of u_1, u_r. Letting $u_{r+1} = z$, we obtain the situation analyzed earlier.

If C has no green vertex, we can still proceed as above. Since C is finite, we must eventually reach the situation described above that includes the triple $(u_{r+1}, \{v_r, u_1\})$. Since u_1 is still only green, we must be able to proceed similarly in the opposite direction, finding triples $(u_1, \{v_1, v_0\})$, $(v_0, \{u_0, u_1\})$, $(u_0, \{v_0, v_{-1}\})$, etc. However, this time, we do not have the option of reaching the corresponding situation where an already enumerated vertex can be used, since all enumerated vertices are already red-green. We also cannot stop with a green vertex, since green vertices were assumed to not exist. This contradicts the finiteness of C. $\square$

A vertex of G is blue if it belongs to no triple of G. Note that a vertex which is red or green in a triple is nonblue in G. Note moreover that a vertex that is green in a triple component C remains green in G, because Lemma 2.1 ensures that it cannot be red in a triple not in C.

Theorem 2.2 implies a result of Sumner [12] that *every point determining graph G has a vertex x such that $G - x$ is point determining.* Indeed, either G has no triples, and thus each $G - x$ is point determining, or there are triples, and triple components, and hence at least one green vertex v in a triple component. As we have observed above a vertex green in one triple component remains green in G, so the graph $G - v$ is point determining.

Recall that Sumner [12] defined the nucleus of a point determining graph G to consist of all vertices v such that $G-v$ is point determining, and showed the following stronger result, which also follows from the above analysis.

Corollary 2.3 *If G is a point determining graph without isolated vertices, then the nucleus of G has at least two vertices.*

Proof. Suppose G is a point determining graph without isolated vertices. It at least two vertices of G belong to no triples of G, then these vertices lie in the nucleus and we are done. If there are at least two vertices not red in any triple, then these vertices lie in the nucleus and we are also done. Note that type-one and type-two triple components have at least two green vertices each, which remain green in G, as discussed above. Thus it remains to consider the situation where one vertex x is green in each triple component containing it, and each vertex belongs to some triple component. If x is green in a triple component C, then it is not adjacent to any vertex of C. Thus x has no neighbours in G, i.e., is an isolated vertex. $\square$

Note that a point determining graph can have at most one isolated vertex.

In what follows, we shall verify that each realization G of a point determining graph H either has no blue vertices, or has all vertices blue. In the latter case G has no triples, and we have each $G - x$ isomorphic to H; we call such a G a *strict realization* of H. In the former case every vertex of G belongs to a triple, and a triple component; we call such a G a *colourful realization.* Thus every realization is either strict or colourful.

Theorem 2.4 *Let H be a point determining graph. There are at most two nonisomorphic realizations of H; at most one is a strict realization, and at most two are colourful realizations.*

All five possible situations allowed by Theorem 2.4 do occur.

- The graph H consisting of two adjacent vertices u, v, has both kinds of realizations G: if the added vertex x is adjacent to both u, and v then G is a strict realization; if the added vertex x is adjacent to neither of u, v, then G is a colourful realization.

- The graph H consisting of a triangle uvw has only the strict realization G in which the added vertex x is adjacent to all three vertices u, v, w.

- The graph H with five vertices forming a path $uvwt$ plus an isolated vertex s only has a colourful realization G, in which the added vertex x adjacent to u, w, s.

- The graph H with the seven vertices s, t, u, v, w, y, z and the eight edges $sy, sz, ty, tz, uw, vw, tv, wz$ has two realizations G, both colourful: one has the added vertex x adjacent to u, and v, and the other has the added vertex x adjacent to u, v, y, and z.

- The graph H with four vertices forming a triangle uvw plus an edge ut has no realization G.

The Theorem will be proved by a sequence of Lemmas which take up the remainder of this Section.

Lemma 2.5 *Each point determining graph has at most one strict realization.*

Proof. Assume H is point determining and G is a strict realization of H. For every vertex x in G, we have that $G - x$ is point determining and thus isomorphic to H. This implies that all vertices of G have the same degree, since if x, x' have different degrees then $G - x, G - x'$ are not isomorphic. Thus either all vertices of H have the same degree d, or the degrees of the vertices of H are d and $d + 1$, for some d. In the latter case, if $G - x = H$, then x must be adjacent precisely to the vertices of degree d, defining G uniquely.

If all vertices of H have degree d, and $G - x = H$, then x is either adjacent to all or none of the vertices of H. If x is adjacent to no vertex of

H, then x has degree zero, and thus all vertices of G have degree zero, and any two vertices of G have the same neighbours, contrary to G being point determining. Thus x is adjacent to all vertices of H, defining G uniquely. $\square$

Lemma 2.6 *If G is a realization of some graph H, then G cannot have a type-two triple component.*

Proof. A type-two triple component C has green vertices u_1, u_{r+1}, so $G - u_1$ and $G - u_{r+1}$ are point determining and thus isomorphic. This means that u_1 and u_{r+1} have the same degree in G. On the other hand, in a type-two triple component C, the difference between the degree of u_{r+1} and the degree of u_1 is r, a contradiction. $\square$

Consider a point determining graph G, and a green vertex x of G. The graph $G - x$ contains all the triples of G that do not involve x; but it may contain new triples which were not triples in G because of x. We first observe that under certain conditions, this does not happen.

Lemma 2.7 *Suppose G is a realization of H. Suppose further that x is the green vertex u_1 of a small type-three triple component C of G. Then all the triples of H are also triples of G.*

Proof. Note that $G - x$ is isomorphic to H, as x is green in C, and hence in G. Since both $(u_2, \{x, v_1\})$ and $(v_1, \{x, u_2\})$ are triples of G, the adjacent vertices u_2, v_1 are similar in G. Consider a triple $T = (y, \{z, t\})$ of H which is not a triple of G. This must mean that x is adjacent to one of z, t but not the other. If z and t are both different from u_2 and v_1, then both of u_2, v_1 have the same adjacencies to z and t as x, hence T is not a triple of H. On the other hand, z, t cannot be the vertices u_2, v_1, since x is nonadjacent to both. Thus we may assume that, say, $u_2 = t$ and $z \neq v_1$. It follows that $y = v_1$, since u_2 is nonadjacent to z and hence v_1 is nonadjacent to z (but adjacent to $u_2 = t$). This contradicts Lemma 2.1 since the triples $(y, \{z, t\}) = (v_1, \{z, u_2\})$ and $(v_1, \{x, u_2\})$ share two vertices, yet v_1 is red in each triple. $\square$

Even if new triples do arise in $G - x$, we may have a handle on where they are. Suppose C is a triple component of G in which x is green. The vertices of $C - x$ are in a triple component C' of $G - x$. Depending on the type of C', there is not much choice where the additional triples of C' may be.

Lemma 2.8 *Suppose G is a realization of H. Suppose further that x is a green vertex of a triple component C of G. Let $(y, \{z, x\})$ be a triple of C. Let C' be the triple component of $G - x$ containing $C - x$.*

Let T be any triple of H which is not a triple of G. Then

- *T contains one of y, z.*
- *If T is in C', then C' is a type-three triple component, and $T = (u_{r+1}, \{v_r, u_1\})$ with $y = u_1, z = v_1$.*
- *If T is not in C', then either*
  - *$T = (t, \{w, y\})$ with y green in H and t not a vertex of C'; the vertex w is either not in C' or it is the other green vertex of C'; or*
  - *$T = (z, \{t', w'\})$ with z red in H, and C' is a small type-two triple component; neither t' nor w' is in C'.*

Proof. We have again $G - x$ isomorphic to H. Depending on what type of triple component C was, and where $(y, \{z, x\})$ lied in it, the triples of C' which belong to $C - x$ form either a type-one component of $G - x$ with $y = u_1$ and $z = v_1$, or a type-two component of $G - x$, with either $y = u_1$ and $z = v_1$, or with $y = u_{r+1}$ and $z = v_r$. (Specifically, if C was a type-one triple component of G, then $C - x$ is a type-two triple component of $G - x$, and if C was a type-three triple component of G, then $C - x$ is type-one triple component of $G - x$.)

If $T = (u, \{v, w\})$ does not contain one of y, z, then since x distinguishes v and w in G, we see that z distinguishes v, w, so T is not a triple of H, a contradiction.

Suppose T is in C'. As in the proof of Theorem 2.2, we analyze where T lies with respect to the triples of $C - x$. We first consider the case where T lies exactly where the triple $(y, \{z, x\})$ lied before x was removed.

If $T = (y, \{z, x'\})$, then z and x' are not adjacent, and hence x and x' are not adjacent, since $(y, \{z, x\})$ is a triple of G. As z and x are also nonadjacent, we have T a triple of G, a contradiction.

If $C - x$ is type-one, then either we have the above situation where $T = (y, \{z, x'\})$, or $T = (v_r, \{u_r, x'\})$ with x' not in $C - x$, which contradicts the fact that T contains y or z.

If $C - x$ is type-two, we again have two places where T can lie.

Without loss of generality, we may assume that $y = u_1$ and $z = v_1$. Then the two places for T are $T = (y, \{z, x'\})$, treated above, or $T =$

$(u_{r+1}, \{v_r, x'\})$, which implies either $x' = u_1$ or $r = 1$. (Otherwise we would again have T not containing y or z.) If $x' = u_1$, then C' is a type-three triple component as claimed. If $r = 1$, then $N(x) \cup \{x\} = N(x') \cup \{x', u_1, u_2\}$ with $N(x') \cap \{u_1, u_2\} = \emptyset$, and x' is green or x', u_{r+1} belong to a type-one triple component D of G. If x' is green, then $G - x$ and $G - x'$ must be isomorphic, which is not possible since the degrees of x and x' differ by two. Otherwise we obtain a contradiction with the difference of the degrees of x and the other green vertex of the type-one component D.

Suppose T is not in C'. If T is attached at y, then Theorem 2.2 implies that y is green and $T = (t, \{w, y\})$; in this case w may be the other green vertex of C', but t is not in C'. If T is attached at z, then Theorem 2.2 implies that z is red, C' is a small type-two triple component, and $T = (z, \{t', w'\})$, with neither of t', w' in C'. □

Lemma 2.9 *If G is a realization of H and G contains a triple component with at least four vertices, then G is a strict or a colourful realization.*

Proof. We shall show that G cannot contain both blue and nonblue vertices. Assume there is a triple component C; let x be a green vertex of C, and let C' be the component of $G - x$ containing the triples of G that do not involve x.

Suppose T is a triple of H which is not a triple of G, and which does not to belong C'. Then T either forms a three-vertex triple component of H, or $T = (t, \{w, y\})$ where t, w belong to a type-one triple component C'' of G not containing x, with $2r$ vertices. The T together with C'' form a triple component C''' of H, with $2r + 1$ vertices.

If C is a triple component of G with at least four vertices, then $G - x$ has strictly fewer vertices belonging to triple components with at least four vertices than G. On the other hand, if x' is chosen blue, then $G - x'$ has all the vertices belonging to triple components with at least four vertices from G, and thus $G - x'$ has at least as many vertices belonging to triple components with at least four vertices as G. Thus $G - x$ and $G - x'$ are both point determining and not isomorphic, a contradiction. □

Lemma 2.10 *If some realization of H has a small type-three triple component, then there is at most one realization of H which is not strict, and that realization must be colourful. In particular, there are at most two realizations of H.*

Proof. Suppose G is a realization of H with a small type-three triple component C. Let x be the vertex identified in Lemma 2.7, and let y and z be the other two vertices of C. The Lemma implies that the vertices y, z are blue in H, with $N(y) \cup \{y\} = N(z) \cup \{z\} = N(x) \cup \{y, z\}$ in G. If another realization G' has a green vertex x', with $G' - x' = H$, then G' does not contain the triple $(y, \{x', z\})$, else G' is isomorphic to G by mapping x' to x. Suppose G' has a triple T containing exactly one of y, z, either $(y, \{x', t\})$ or $(u, \{x', y\})$. If $T = (y, \{x', t\})$ and there is a triple $T' = (t, \{x', y\})$, then G' is isomorphic to G by permuting the vertices t, y, z. Otherwise, if $T = (y, \{x', t\})$ then x' and t are both green by Lemma 2.7 and have different degrees in G', thus G' is not a realization of H. Similarly, if $T = (u, \{x', y\})$ then x' and y are both green and have different degrees, so G' is not a realization of H.

Thus both y and z are blue in G', and by Lemma 2.9 G' may not contain a triple component with at least four vertices. Then there is a small triple component involving x' in G', and by Lemma 2.7 $G' - x'$ has fewer vertices belonging to triple components than G'. On the other hand, $G' - y$ has at least as many vertices belonging to triple components as G'. So $G' - x'$ is not isomorphic to $G' - y$, and G' is not a realization of H. Therefore the only possible other realization G' of H has all vertices blue and is unique by Lemma 2.5. □

Lemma 2.11 *If H has two adjacent similar vertices then H has at most two realizations.*

In particular, if H has a small type-three triple component then it has at most two realizations.

Proof. Suppose y and z are similar in H. No triple of H can contain exactly one of y, z, because we could obtain another triple by exchanging y and z, contrary to Lemma 2.1. Therefore either y, z are blue in H, or there is a vertex t in H such that $(y, \{t, z\})$ and $(z, \{t, y\})$ form a triple component in H and these are the only triples in H involving either y or z. This implies that each of y, z is either blue in G or belongs to a triple component with exactly three vertices in G. Indeed, otherwise x, y, z, t would form a triple component with four vertices in G with x and t similar and nonadjacent, contrary to G being point determining. If G has a triple component with three vertices, then the Lemma follows from Lemma 2.10. Otherwise y, z are both blue and the Lemma follows from Lemmas 2.5 and 2.9. The second part of the lemma comes from the fact that a type-three triple component of H has the adjacent vertices u_2 and v_1 similar in H. □

Lemma 2.12 *If H has no small triple component, then H has at most two realizations.*

Proof. For each realization G of H which is not strict, select a green vertex $v = v_G$ which minimizes the maximum number $r = r_G$ of vertices in a triple component C_G containing v. Select a non-strict realization G' of H with the largest $r_{G'}$. The graph $G - v_G = H$ has vertices u_G, w_G such that $(u_G, \{v_G, w_G\})$ is a triple in C_G. We claim that $u_{G'} \neq u_G$ for all other non-strict realizations G of H. Suppose $u_G = u_{G'}$, and consider the corresponding triple components $C_G - v_G$ and $C_{G'} - v_{G'}$ in H. The triple component $C_G - v_G$ cannot be a triple component in G', else $C_{G'}$ would share the red-green vertex $u_{G'}$ with another triple component. Thus in Lemma 2.8 with $C' = C_{G'} - v_{G'}$ there is a triple T not in C' that belongs to $C_G - v_G$, where $T = (t, \{w, u_G\})$ and t, w belong to a type-one triple component C'' of G' with $2r$ vertices, not containing $v_{G'}$. Then we obtain the triple component $C_G - v_G$ in H by adding u_G, and $C_G - v_G$ has $2r + 1$ vertices. The triple component $C_{G'}$ has $r_{G'}$ vertices, and the triple component C_G has r_G vertices, with $r_{G'} \geq r_G$. Thus C'' is a triple component of G with $r_G - 2$ vertices and green vertex t that belongs only to the triple component C''. We could thus set $v_{G'} = t$ and achieve $r_{G'} = r_G - 2 \leq r_{G'} - 2$, a contradiction.

Thus $u_{G'} \neq u_G, u_{G''}$ for any two non-strict realizations G, G'' not isomorphic to G'. If $u_G = u_{G''}$, then say $r_{G''} \geq r_G$, and by the preceding argument we could set $r_{G''} = r_G - 2 \leq r_{G''} - 2$, a contradiction. Therefore if G, G', G'' are not isomorphic, then the three vertices $u_G, u_{G'}, u_{G''}$ are distinct. Note that if in going from $H = G - x$ to G the vertex $u_{G'}$ ceases to be green, then G has a type-three triple component D containing both x and $u_{G'}$ and $D - x$ is the only triple component of H containing $u_{G'}$, whence G and G' would be isomorphic.

If C_G is a type-one triple component, with the edge-set $u_i v_j$ with $j < i$, then v_r has degree zero in C_G and $u_G = v_1$ has degree $r - 1$ in $C_G - v_G = H$, a decrease in degree $r - 1$. If C_G is type-one triple component with the edge-set $u_i v_j$ with $j \geq i$, then v_r has degree r in C_G and $u_G = v_1$ has degree zero in $C_G - v_G = H$, an increase of r in the degree. If C_G is a type-three triple component, then $v_G = u_1$ has degree zero in C_G and $u_G = v_1$ or $u_G = u_{r+1}$ has degree r in $C_G - v_G = H$, a decrease of r in the degree. On the other hand, in going from $G - v_G$ to G, the degree of $u_{G'}$ stays the same or increases by one. Say the degrees of u_G and $u_{G'}$ in $G - v_G$ satisfy $d_G = d_{G'}$. Then in going from $G - v_G$ to G we have a degree $d_{G'}$ that stays the same or goes up by one, while for the degree d_G we have a decrease of at least one or an increase of at least two. This gives two green vertices x, y with different degrees in G, contrary to the fact that $G - x$ and $G - y$ are isomorphic. Otherwise say $d_{G'} < d_G < d_{G''}$, so $d_{G''} \geq d_{G'} + 2$, and in going from $G - v_G$ to G the degrees $d_{G'}, d_{G''}$ stay the same or increase by one, giving a again two green vertices x, y with different degrees in G, contrary to the fact that $G - x$ and $G - y$ are isomorphic.

By Lemma 2.5, there are at most two non-strict realizations G of H, and at most one strict realization G, for at total of at most three realizations G. Suppose there is a strict realization graph G''. Then as in Lemma 2.5 the degrees in H cannot differ by more than one. In Theorem 2.2, vertices u_1 and u_3 have degrees that differ by two, so H cannot have triple components with more than four vertices, and so by the assumption of the Lemma all triple components of H have four vertices, and are type-one components, with $r = 2$. As before, we consider the vertices u_G and $u_{G'}$ of degrees d_G and $d_{G'}$ respectively, which now must satisfy $d_G \leq d_{G'} + 1$. We must C_G a type-three component, with $2r + 1 = 5$ vertices, so in going from $C_G - v_G$ to C_G we have a decrease in degree of $r = 2$, giving degree at most $d_G - 2 \leq d_{G'} - 1$, while the degree of $u_{G'}$ stays the same or goes up by one, giving degree at least $d_{G'}$. This would mean not all green vertices of G have the same degree, a contradiction. Therefore if there is a strict realization G'' of H, then there is at most one other realization G of H. This shows that there are at most two realizations G. □

Lemma 2.13 *If H has a small type-two triple component, but no small type-three triple component, then H has at most two realizations.*

Proof. There is at most one strict realization, and all other realizations must be colourful by Lemmas 2.9 and 2.10. The small type-two triple component in H consists of a triple $(v_1', \{u_1', u_2'\})$. If $G - x = H$ with x green, then v_1' will become blue in G unless v_1' belongs to a type-one triple component C of G with $r = 2$, with vertices u_1, u_2, v_1, v_2, having $x = v_2$ and $v_1 = v_1'$. In particular, $C - x$ is the triple $(v_1, \{u_1, u_2\})$. If vertex u_2 belongs to another triple component D in H, then D has an odd number $2r + 1$ of vertices, and adding x to H (to obtain G) we remove u_2 and a triple containing u_2 from the triple component D. Assume x is chosen in G to belong to a single triple component C with four vertices, if this is possible.

Suppose there is such a D with $2r+1 \geq 5$ vertices. Then as we go from G to H, the triple component D becomes $D - u_2$, with four vertices, and with green vertices belonging only to the triple component $D - u_2$. Thus x and G

satisfy the assumed property that x belongs to a single triple component C with four vertices. Suppose there is another small type-two triple component with the triple $(v_1, \{u'_1, u'_2\})$ containing v_1 in H, and with G', x' such that G' contains a triple component C', where $C' - x'$ is $(v_1, \{u'_1, u'_2\})$, as for $G, x, C, (v_1, \{u_1, u_2\})$. The green vertex u'_2 in H will become blue in G unless u'_2 belongs to a triple component D' of size $2r'+1 \geq 5$ in H and in G. Then $G - u'_2$ is isomorphic to H, so by Lemma 2.8 the two green vertices of $D' - u'_2$ must meet one of the two triples $(v_1, \{u_1, u_2\})$, $(v_1, \{u'_1, u'_2\})$ under the isomorphism between $G - u'_2$ and H. Indeed, one of these two triples is removed in going from H (isomorphic to $G - u'_2$) to G. But they cannot both meet $(v_1, \{u_1, u_2\})$ at u_1, u_2 respectively, since $D' - u_2$ has an even number $2r \geq 4$ of vertices, while u_2 meets only triple components with an odd number of vertices, and similarly they cannot both meet $(v_1, \{u'_1, u'_2\})$. Thus there are at most two graphs G, G' and they must have $u'_1 = u_2$ and $u'_2 = u_1$, or else $2r + 1 = 2r' + 1 = 3$.

Suppose $2r + 1 = 2r' + 1 = 3$. This means that G and G' have triple components C and C' of size four so that $C - x$ and $C' - x'$ meet u'_2 and u_2 respectively. If there is another small type-two triple component with triple $(v_1, \{u''_1, u''_2\})$ in H, then the vertex y that is red-green in G and green in H as in Lemma 2.8 must meet both $(v_1, \{u'_1, u'_2\})$ and $(v_1, \{u''_1, u''_2\})$, which is not possible. Thus $(v_1, \{u_1, u_2\})$ and $(v_1, \{u'_1, u'_2\})$ are the only triples of H containing v_1. If there is a third realization G'', then it must have a corresponding small type-two triple component with triple $(v_1, \{u''_1, u''_2\})$, where, say, $u''_1 = u'_2, u''_2 = u'_1$, and G must have a triple component C''' of size four so that $C''' - x$ meets $u''_2 = u'_1$. But then by Lemma 2.8 $C' - x'$ must meet both $C - x$ and $C''' - x$ at the red vertex, since $C' - x'$ meets $(v_1, \{u_1, u_2\})$ at a green vertex. This is not possible as C and C''' can share only green vertices.

We have thus shown that there are at most two colourful realizations G and G'. Suppose there is also a strict realization G''. Then as in the proof of Lemma 2.5 the degrees in H differ by at most one, and every triple component in H has at most four vertices, thus is type-one with $r = 2$. Suppose first that $u_1 = u'_2$ and $u_2 = u'_1$, with u_1 of degree d and u_2 of degree $d + 1$. Then in going from $G - H$ to G we remain at degree $d_G = d$ for green vertices, while in going from $G' - H$ to G' the degree becomes $d_{G'} = d + 2$ for green vertices. (This is easily seen by considering the two possible cases for the edge-set of a triple component of size four.) Thus all triples $(v_1, \{u_1, u_2\})$ that belong to triple components of size four in G must have the same d in H, and the only edge in such triples is the edge (v_1, u_2). Therefore when we add $x = v_2$, we have x nonadjacent to any of

v_1, u_1, u_2, or to any other vertex in a type-three triple component of size five containing x. Since G has no blue vertices, we have that H must contain some triple component F with four vertices that remains a triple component in G. Otherwise $x = v_2$ is isolated in G, and u_1 must also have degree zero and be isolated in G. This means that u_1 and v_2 are similar in G, contrary to G being point determining. The degree of a green vertex in F stays the same or goes up by one from H to G. On the other hand, the degree of a green vertex either stays the same or goes up by one from H to G', or goes down by two from H to G'. (The latter case occurs when F becomes a component with five vertices for the new green vertex x' in this component.) Thus $d_{G'} \leq d_G + 1$, contrary to $d_{G'} = d_G + 2$. Thus in this case there is at most one such G without blue vertices, and therefore at most two graphs satisfying Theorem 2.4.

In the remaining case, the situation $u_1 = u_2'$ and $u_2 = u_1'$ does not arise for any triple. Let A be the set of small type-two triple components, with triples $(v_1, \{u_1, u_2\})$, in H, that lead to a triple component of size four with x. Let A' be the set of small type-two triple components, with triples, $(v_1', \{u_1', u_2'\})$, in H, that lead to a triple component of size four with x'. Every triple in A must meet a triple in A' at v_1 or u_2, and every triple in A' must meet a triple in A at v_1' or u_2'. Thus for any such pair of triples in A and A' we must have $v_1 = v_1'$ or $u_2 = u_2'$. Furthermore, a triple in A cannot meet two triples in A' at v_1, because these two triples would then have the same v_1'. Similarly, a triple in A cannot meet two triples in A' at u_2, because these two triples would have the same u_2'. There are thus at most two triple components in A and at most two triple components in A'. As observed earlier, this case cannot happen for $2r + 1 = 5$, so $2r + 1 = 2r' + 1 = 3$, whence A and A' contain exactly two triples each.

Thus suppose A consists of $(v_1, \{u_1, u_2\})$ and $(v_1'', \{u_1'', u_2''\})$, and B consists of $(v_1', \{u_1', u_2'\})$ and $(v_1''', \{u_1''', u_2'''\})$, with $v_1 = v_1'$, $v_1'' = v_1'''$, $u_2 = u_2'''$, $u_2' = u_2''$. Then x and x' each belong to exactly two triple components of size four, because $G - x = G' - x' = H$. Since $G - z = H$ for every green vertex z of G, we have that every green vertex z of G belongs to exactly two triple components of size four. Similarly every green vertex z' of G' belongs to exactly two triple components of size four. If G has a triple component E of size five and x is chosen in it, then either of the two green vertices in $E - x$ could be the vertex y in Lemma 2.8 that becomes red-green in G, so both green vertices in $E - x$ have to meet the two triples in A', which is not possible since the two triples in A' share at most one vertex. Thus G and G' have no triple component of size five. If $u_1 = u_1''$, then u_1 belongs to two triple components of size four in G arising from the two triples in

A, so u_1 belongs to no triple component of size four in H to maintain the total of two triple components of size four for z green in G, and therefore u_1 would necessarily be blue in G', contrary to the assumption. If $u_1 \neq u_1''$, then u_1 belongs to one triple component of size four in G arising from one triple in A, so u_1 must belong to one triple component of size four in H to maintain the total of two for z green in G, and therefore u_1 would belong to just one triple component of size four in G', contrary to shown for z' green in G'. Therefore in this case there is at most one colourful realization G, and therefore at most two realizations. □

Lemmas 2.11, 2.12 and 2.13 complete the proof of Theorem 2.4. We now suggest that the proofs can be interpreted as a polynomial time algorithm which, given a point determining graph H, outputs two (or fewer) graphs G which are the only possible candidates foe realizations of H. The algorithm does not certify that either of these at most two graphs G actually is a realization of H. (This would seem to require a difficult isomorphism testing between various subgraphs $G - x$ and H.)

The algorithm proceeds first to seek a strict realization G - as long as the vertices of H have degrees that differ at most one. In that case it follows the proof of Lemma 2.5, and results in at most one such graph G, according to the proof. Then it attempts to find colourful realizations, by following the proofs of Lemmas 2.9 and 2.10. If there are two similar vertices y, z in H, then the algorithm produces at most one graph G containing a small type-three triple component having vertices x, y, z, following the proofs of Lemmas 2.10 and 2.11, and then the algorithm halts. Otherwise no potential realization G has a small type-three triple component. If H has no small type-two triple component, then the algorithm produces at most two graphs G obtained by adding x and such that x is a vertex in G that minimizes the largest triple component to which it belongs, following the proof of Lemma 2.12; or it produces at most one realization G if the degrees of H differ by at most one, and the algorithm halts. In the final case where H has a small type-two triple component, the algorithm produces at most two realizations G by following the proof of Lemma 2.13, or at most one realization G if the degrees of H differ by at most one.

3 Obstructions to Full Homomorphisms

In this section, we will generally consider graphs H with loops allowed (while graphs G will have no loops). However, to begin we focus on graphs H also without loops. Recall that our terminology employs the term *graph* if loops are disallowed, and the term *graphs with loops allowed* otherwise.

Amongst graphs, the composition of full homomorphisms is defined, and easily seen to be a full homomorphism. (For graphs with loops allowed, this is no longer the case, at least not with our definitions.) A full homomorphism f of a graph G to a graph H can identify only nonadjacent similar vertices (in other words, if $f(x) = f(y)$ then x, y are nonadjacent similar vertices of G). Thus a full homomorphism of a point determining graph G to a graph H must be injective, i.e., an isomorphism onto an induced subgraph of H. In particular, H has at least as many vertices as G. It also follows that every graph G contains a unique (up to isomorphism) minimal subgraph G^* to which it admits a full homomorphism; here G^* is obtained from G by identifying all nonadjacent similar vertices. Thus G is obtained from G^* by replacing each vertex x by an independent set S_x of vertices, with a vertex in S_x adjacent to a vertex in S_y if and only if x is adjacent to y in G^*. Note that G^* is a point determining subgraph of G. It also follows that G admits a full H-colouring if and only if G^* admits a full H^*-colouring. In other words, deciding if there exists a full homomorphism between two graphs G and H reduces to deciding this for point determining graphs G and H. In particular, we may assume H is point determining, and conclude that a minimal H-obstruction must be point determining.

Theorem 3.1 *If H is a graph with k vertices, then*

1. *each minimal H-obstruction has at most $k + 1$ vertices, and*

2. *there are at most two minimal H-obstructions with $k + 1$ vertices.*

Proof. We first prove the claim 1, by elementary means. As explained above, it will suffice to prove the theorem for point determining graphs H, since the point determining graph H^* has at most k vertices. Moreover, a minimal H-obstruction G is point determining. As noted above, if G has more than k vertices, it does not admit a full homomorphism to H. According to Corollary 2.3, G contains a point determining subgraph of each smaller size. Thus the minimality of G implies that it has at most $k + 1$ vertices.

To prove the claim 2, we shall need to invoke the results of our analysis of point determining graphs. The surprizing connection is stated as the next lemma, which implies that claim 2 also holds. □

Here is the connection between minimal obstructions and realizations that is the crux of our paper.

Lemma 3.2 *Let G and H be graphs with $k+1$ and k vertices respectively. Then G is a minimal H-obstruction if and only if it is a realization of H.*

Proof. We may assume both G and H are point determining, as discussed above.

If G is a minimal H-obstruction, then each $G - x$ admits a full H-colouring, so in those cases when $G - x$ is point determining this is an isomorphism of $G - x$ and H.

On the other hand, suppose G is a realization of H. There is no full H-colouring of G, as G has more vertices than H. If G is a strict realization, all $G - x$ are isomorphic to H, thus admit a homomorphism to H, and G is a minimal H-obstruction. If x is a green vertex of G, then $G - x$ is point determining, thus isomorphic to H, and so $G - x$ admits a full H-colouring. If x is red in some triple component C, in a triple (x, y, z) of C such that y is green in C (and hence in G), then we observe that $G - x$ admits a full H-colouring if and only if $G - x - y$ admits a full H-colouring, since y and z can have the same image. Since $G - y$ admits a full H-colouring by the above argument, it follows that so does $G - x - y$ and hence $G - x$. If there are two triples, say, $(x, \{y, z\})$ and $(y, \{x, t\})$, between x and a green vertex t of C, we argue that $G - t$ admits a full H-colouring, thus $G - t - y$ and therefore also $G - y$ also admit a full H-colouring. Now $G - y - x$ and thus also $G - x$ admit a full H-colouring. Let s denote the number of triples of C between x and a green vertex of C. The argument described above for $s = 1$ translates in the obvious way to a proof, by induction on s, of the assertion that each $G - x$ admits a full H-colouring. Thus G is a minimal H-obstruction. □

We now begin to consider graphs H with loops allowed. Recall that the complement $\overline{G}$ of a graph G has ij an edge if and only if ij is not an edge of G. For graphs H with loops allowed, we use the same definition, including the case $i = j$: thus loops in H become nonloops in $\overline{H}$ and vice versa. Let G be a graph and let H be a graph with loops allowed. It is easy to see that G admits a full H-colouring if and only $\overline{G}$ admits a full $\overline{H}$-colouring.

Focus first on the case of graphs H in which every vertex has a loop.

Corollary 3.3 *If H is a graph with loops on all its ℓ vertices, then each minimal H-obstruction has at most $\ell + 1$ vertices, and there are at most two minimal H-obstructions with exactly $\ell + 1$ vertices.*

Proof. This follows from the Theorem by complementation. □

In the general case of graphs H with loops allowed, we may now focus on graphs H in which there is at least one vertex without a loop and at least one vertex with a loop.

We first construct a family of graphs with $k > 0$ vertices without loops and $\ell > 0$ vertices with loops. We begin with any vertex-transitive graph $Z = Z_k$ with $k+1$ vertices. (A graph Z is *vertex-transitive* if for any two vertices u, v of Z there is an automorphism of Z taking u to v; cycles and complete graphs are obvious examples.) We denote by $H = H_{k,\ell}(Z)$ the graph with loops allowed, obtained from Z by replacing one vertex v of Z by a set of ℓ loops, with no edges between them. (Each of these vertices is adjacent in H to exactly the same vertices as v was in Z.) Now H has k vertices without loops, and ℓ vertices with loops. Let $G = G_{k,\ell}(Z)$ be the graph obtained from Z by replacing *each* vertex of Z by $\ell + 1$ independent vertices. It is easy to see that G does not admit a full homomorphism to H, because two nonadjacent vertices cannot map to the same vertex with a loop, but if any vertex x is deleted, then $G - x$ does admit a full homomorphism to H. In other words, G is a minimal H-obstruction; of course this means that $\overline{G}$ is a minimal $\overline{H}$-obstruction. Note that G has $(k+1)(\ell+1)$ vertices.

The main result for graphs with loops allowed can be stated as follows.

Theorem 3.4 *Let H be a graph with $k > 0$ vertices without loops and $\ell > 0$ vertices with loops. Then*

1. *every minimal H-obstruction has at most $(k+1)(\ell+1)$ vertices,*
2. *there is at most one minimal H-obstruction with $(k+1)(\ell+1)$ vertices, and*
3. *this only occurs when H is $H_{k,\ell}(Z)$ or its complement, for some Z; in this case the unique minimal H-obstruction is $G = G_{k,\ell}(Z)$ or its complement.*

Proof. Recall that vertices u and v of a graph G are *similar* in G, if they have exactly the same set of neighbours, other than u and v, in G. Similar vertices may be adjacent or nonadjacent; we have already noted that similarity is an equivalence relation on the vertices of G, and that each equivalence class induces an independent set or a clique. Let G be a minimal H-obstruction, and let $S_1, S_2, \ldots, S_r$ be the equivalence classes of the relation of similarity in G.

The case $k = \ell = 1$ is easily seen to have two minimal H-obstructions: If H consists of an isolated loop and an isolated nonloop, then the minimal

H-obstructions are $2K_2$ and the path with three vertices; otherwise H is the complementary graph with loops allowed, and the minimal H-obstructions are the complements of the above two graphs. Thus in the sequel we shall assume that $k + \ell \geq 3$.

We first consider the case that $r \geq k + \ell + 1$.

A graph G with $r \geq k + \ell + 1$ similarity classes cannot admit a full homomorphism to H, since nonsimilar vertices in G cannot map to the same vertex of H. If a similarity class S_i has at least three vertices, we can remove one vertex and maintain the same similarity classes. (This is not so for classes with two elements, as the removed vertex could have been the only vertex distinguishing its mate from some other vertex.) This means that the minimal H-obstruction G has at most $2r \leq 2k + 2\ell + 2$ vertices. The remainder of the proof in this case consists of a technical case analysis proving that in fact the bound can be improved from $2k + 2\ell + 2$ to $(k + 1)(\ell + 1)$.

Select one vertex v_i from each S_i. We first deal with the possibility that they induce a graph with one similarity class. If the vertices v_i are independent, then there are no edges of G joining the sets S_i, and hence at least $k + \ell$ of them are cliques of size at least two. Now G is not a minimal H-obstruction because it properly contains the obstruction induced by $k+\ell$ sets S_i that are cliques of size at least two. Vertices in different cliques S_i must map to different vertices of H because they are not similar, and no clique of size at least two can be mapped to a vertex of H without a loop. This is impossible as H as exactly $k + \ell$ vertices, at least one of which has no loop. A similar argument applies if the vertices v_i form a clique.

If the r vertices v_i form neither an independent set nor a clique, then we may choose $k+\ell+1 \leq r$ of these vertices that form neither an independent set nor a clique. Consider the subgraph G' of G induced by these $k + \ell + 1$ vertices v_i. We have ensured that the relation of similarity in G' has at least two equivalence classes. The vertices v_i were all pairwise nonsimilar in G, thus we may add to G' some other vertices $u_1, \ldots, u_{t-1}$, $t \leq k + \ell$, of G, distinguishing the vertices v_i, and thus forming a graph G'' with at least $k + \ell + 1$ similarity classes. (We can do this by repeatedly adding a vertex u_j from G distinguishing two currently similar vertices v_i.) The graph G'' is therefore an obstruction of size $2t = 2(k + \ell) \leq (k + 1)(\ell + 1)$.

For the proof of statement 3, we also analyze when the equality occurs. In fact, we will show that the size of this obstruction may be reduced to $2t - 1 < (k + 1)(\ell + 1)$. If there are at least three equivalence classes under similarity in G', then we need only choose $t-2$ vertices u_j. Thus we assume G' has exactly two equivalence classes X and Y, with $x = |X|$ and $y = |Y|$ satisfying $x, y \geq 1$ and $x + y = t + 1$. We only need to choose $t - 1$ vertices u_j if no u_j distinguishes more than one pair of currently similar vertices v_i; thus assume that each u_j distinguishes vertices in exactly one of X and Y (in other words, it is completely adjacent or completely nonadjacent to either Y or X). If u_j distinguishes vertices in X (respectively in Y), then we may assume u_j is adjacent to exactly one v_i in X if X is an independent set (and nonadjacent to exactly one v_i in X if X is a clique). Moreover, these corresponding u_j and v_i have identical neighbors in Y. Otherwise u_j would not only subdivide one class but also introduce a new class in the current G', since it is different from the classes of the previous G'. Therefore at most $t - 2$ vertices u_i would be needed. Furthermore, if X is independent, then two u_j corresponding to two v_i in X must be also independent, otherwise we could exchange one u_j and its corresponding neighbour v_i and have the other u_j adjacent to two v_i. Thus the subgraph X' of G'' induced by X and its corresponding vertices u_j, consists of $x - 1$ copies of K_2 and one isolated vertex, or the complement of this graph. Similarly the subgraph Y' of G' induced by Y and its corresponding u_j consists of $y-1$ copies of K_2 and one isolated vertex, or the complement of this graph. Furthermore the vertices of X' and Y' are either all adjacent or all non-adjacent to each other.

The argument then reduces to two cases up to complementation. If X is a clique and Y is an independent set not adjacent to X, we have $x \geq 2$. Let us remove from X' a vertex v_i which has a corresponding u_j (recall that exactly one v_i does not). The graph $G'' - v_i$ then has $t = k + \ell$ similarity classes, and a full homomorphism f to H; the f maps the clique $X - v_i$ to a clique of size $x - 1$ in H and the independent set Y to isolated vertices in H; thus H consists of a clique with $x - 1$ vertices and a set of isolated vertices. On the other hand, when we remove from Y' the vertex $v_{i'}$ which does not have a corresponding u_j, then the graph $G'' - v_{i'}$ has $t = k + \ell$ similarity classes, and a homomorphism f' taking X to a clique in H of size x and taking $Y - v_{i'}$ to isolated vertices in H. This contradicts the form of H derived above.

A similar argument applies in the other case, when both X and Y are independent sets, joined by all edges. When we remove the v_i without a corresponding u_j from either X' or Y', we obtain $t = k + \ell$ similarity classes, but both correspond to the same H only if $x = y$; in this case H is a complete bipartite graph with $x - 1$ vertices with loops on one side and x vertices without loops on the other. On the other hand, when we remove from X' a vertex u_j, we obtain a similar partition into similarity classes, except now the complete bipartite graph H has one of the $x-1$ loops missing.

We now consider the case that $r \leq k+\ell$.

In this case, there is also an easy first bound. If the minimal H-obstruction G has a similarity class S_i with more than $\max(k,\ell)+1$ vertices, then removing a vertex from this class allows a homomorphism f to H. If S_i is a clique, then f must map at least one of the remaining $\max(k,\ell)+1$ vertices of S_i to a vertex v with a loop in H. This means that the removed vertex can also be mapped to v. A similar argument applies if S_i is an independent set. This implies that G has at most $r(\max(k,\ell)+1) \leq (k+\ell)(\max(k,\ell)+1)$ vertices. We now proceed to improve this trivial bound to $(k+1)(\ell+1)$.

Let $T_1,\ldots,T_p$ be the classes S_i that are independent, arranged in order of nondecreasing size, and let $U_1,\ldots,U_q$ be the classes S_i that are cliques, also arranged in order of nondecreasing size. Those classes that consist of a single element may be treated either as cliques or independent sets (but not both). We have $p+q=r\leq k+\ell$. Define sets T'_i consisting of two elements from the corresponding T_i, or the single element if $|T_i|=1$, and sets U'_i consisting of two elements from the corresponding U_i, or the single element if $|U_i|=1$. These chosen elements for T'_i and U'_i already give r different similarity classes, one for each T'_i and U'_i. Clearly the total number of elements in the sets T'_i and U'_i is at most $2r \leq 2(k+\ell) \leq (k+1)(\ell+1)$. We may only have equality if $r=k+\ell$ and all T'_i and U'_i have two elements, in which case these chosen elements form an obstruction, since otherwise we may assign T_i and U_i to H in the same way T'_i and U'_i are assigned, so $|T_i|=|U_i|=2$. In this case we define instead one T'_i or U'_i to be a singleton, so that only $2r-1$ elements are chosen for the sets T'_i and U'_i, while guaranteeing that the T'_i and U'_i give r different similarity classes. This can be guaranteed by first choosing one element from each T_i and U_i, for a total of r elements forming at least one similarity class, and then choosing the remaining $r-1$ elements to obtain r separate similarity classes as before.

We shall gradually add elements from the T_i and U_i to the corresponding T'_i and U'_i, maintaining the same nondecreasing order of the sizes of the sets T'_i and U'_i, and keeping the total number of elements in the union of all the T'_i and U'_i at most $(k+1)(\ell+1)$, until they form an H-obstruction.

Let A denote the subgraph of H induced by the k vertices without loops, and let B denote the subgraph of H induced by the ℓ vertices with loops. For each partition of the collection of sets T_i and U_i into a collection P to map to A and a collection Q to map to B, either (1) P does not map to

A, or (2) Q does not map to B, or (3) P maps to A and Q maps to B, yet P, Q together do not map to A, B. Determine for each such partition $R = (P, Q)$ whether (1) or (2) holds, and choose one or the other if both hold; otherwise (3) holds. Each partition $R = (P, Q)$ has a corresponding partition $R' = (P', Q')$ of the collection of sets T'_i and U'_i, but it may be that neither (1) nor (2) holds for $R' = (P', Q')$, in which case if $R = (P, Q)$ is of type (1) then we need to add elements to sets U'_i in P' until (1) holds for P', and if $R = (P, Q)$ is of type (2) then we need to add elements to sets T'_i in Q' until (2) holds for Q'. We shall show how to do this for type (1) until condition (1) holds for P' while preserving the required conditions; doing this for type (2) until condition (2) holds for Q' follows by the same argument exchanging k and ℓ. The case where (3) holds is treated by adding elements as for both cases (1) and (2).

Given $R' = (P', Q')$, let P'_1 be the sets T_i or P', let P'_2 be the sets U_i of P', let Q'_1 be the sets T_i of Q', and let Q'_2 be the sets U_i of Q'. Let p_1 be the number of sets in P'_1, and let s_1 be the total number of vertices in sets in P'_1. Similarly to how p_1, s_1 are defined for P'_1, define p_2, s_2 for P'_2, define q_1, t_1 for Q'_1, and define q_2, t_2 for Q'_2. Select $R' = (P', Q')$ of type (1) for which neither condition (1) nor (2) holds and such that the U'_i in P' for which i is largest has the smallest value of i. Greedily add one element at a time to sets U'_j with $j \leq i$ while preserving the nondecreasing property until (1) holds. If (1) still does not hold and it is not possible to greedily add more elements, then $|U'_i| < |U_i|$ and $|U'_j| = |U'_i|$ for all $j \geq i$. In this case, add one element to each of the sets $U'_i, U'_{i+1}, \ldots, U'_q$ and go back to the greedy process for $j < i$. Eventually (1) will hold, since we would eventually reach the situation where $U'_j = U_j$ for all $j \leq i$. While (1) does not hold, we have $p_1 + s_2 \leq k$, because if $p_1 + s_2 \geq k + 1$, then the corresponding $p_1 + s_2$ elements, one from each set in P'_1 and all the elements of sets in P'_2, must map to different vertices of A, which is not possible since A has only k vertices. Thus by the time the process is complete we will have $p_1 + s_2 \leq k + 1$. We must also have $q_2 + t_1 \leq \ell$, because if $q_2 + t_1 \geq \ell + 1$ then already condition (2) holds, since we would have $q_2 + t_1$ elements that map to different vertices in B, namely one from each set in Q'_2 and all the elements of the sets Q'_1. This process will never increase the size of a set U'_j beyond $k + 1$, since this would correspond to making $s_2 > k + 1$; similarly, the symmetric process will never increase the size of a set T'_j beyond $\ell + 1$, so $s_1 \leq (\ell + 1)p_1$. Finally, the fact that $R' = (P', Q')$ is chosen in the order with U'_i such that i is smallest implies that $|U'_j| = |U'_i|$ for all $j \geq i$, and since $|U'_i| \leq s_2$, we have $t_2 \leq s_2 q_2$. Also $s_2 \geq p_2 \geq 1$ since the process would otherwise not involve sets U'_i in P'_2.

Combining these four inequalities gives $s_1+s_2+t_1+t_2 \leq (\ell+1)p_1+s_2+t_1+s_2q_2 \leq (\ell+1)p_1+s_2(q_2+1+t_1) \leq (\ell+1)p_1+s_2(\ell+1) \leq (p_1+s_2)(\ell+1) \leq (k+1)(\ell+1)$. This completes the proof of the bound $(k+1)(\ell+1)$.

This inequality holds with equality only if $t_1 = s_2t_1$, that is, only if $t_1 = 0$ or $s_2 = 1$. However, if $s_2 = 1$, then the process would not involve sets U_i' in P_2', since there would be just a single such set and no element would have been added to it. And if $t_1 = 0$, then the inequality holds with equality only if $q_2 = \ell$, so that there are at least $\ell+1$ sets U_i, and only if $s_1 = (\ell+1)p_1$. If we assume that there is at least one set T_i, then this last equality can only happen if type (2) was taken care of before type (1), so by reversing the order in which the two types are taken care of, we infer that there are at least $k+1$ sets T_i. Then the total number of sets T_i and U_i is $r \geq (k+1)+(\ell+1)$, contrary to assumption. Thus there are no sets T_i. This completes the proof of statements 1 and 2.

We now consider when the inequality holds with equality, i.e., when there may exist a minimal H-obstruction with $(k+1)(\ell+1)$ vertices. It is clear that this can happen only if $t_2 = s_2q_2$ and $s_2 = k+1$. This implies that some U_i is a clique of size $k+1$. Consider now the time when the last vertex is added to a clique U_i' of size $k+1$. Since $s_2 \leq k+1$, only one U_i' contributes to s_2, so $p_2 = 1$. Furthermore $q_2 \leq \ell$, so there are exactly $p_2+q_2 = \ell+1$ sets U_i, and the total number of vertices can be $(k+1)(\ell+1)$ only if each set U_i has $k+1$ vertices. Let Z be the subgraph of G induced by $\ell+1$ vertices, one from each U_i. Then Z is the complement of a point determining graph, so there is a vertex v in Z such that $Z-v$ is the complement of a point determining graph. Then $G-v$ has a solution by minimality, and if $v \in U_c$, we have that the solution assigns the k vertices in $U_c - v$ to the k parts in A by a one-to-one correspondence, and assigns the ℓ sets U_j for $j \neq c$ to the ℓ parts in B by a one-to-one correspondence, since $Z-v$ is the complement of a point determining graph. Thus M is matrix associated with the complement of a point determining graph Z_ℓ, and the minimal obstruction with $(k+1)(\ell+1)$ vertices is the complement of $G_{k,\ell}(Z)$. If we consider a vertex $w \in U_i \cap Z$ for some $i \neq c$, then $G-w$ has a solution by minimality which assigns $U_i - w$ to the k parts in A by a one-to-one correspondence. This solution cannot assign two vertices $u_{j'}, u_{j''}$ from two different U_j with $j \neq i$ to the same part in B since Z is the complement of a point determining graph. Then the correspondence between the solutions for $G-v$ and $G-w$ gives an isomorphism of Z mapping v to w, so Z is vertex transitive. $\square$

References

[1] R. N. Ball, J. Nešetřil, and A. Pultr, Dualities in full homomorphisms, manuscript 2006.

[2] R.C. Entringer and L.D. Gassman, Line-critical point determining and point distinguishing graphs, Discr. Math. 10 (1974) 43–55.

[3] T. Feder and P. Hell, Matrix partitions of perfect graphs, Discrete Applied Math., to appear.

[4] T. Feder, P. Hell, S. Klein, and R. Motwani, Complexity of list partitions, Proc. 31st Ann. ACM Symp. on Theory of Computing (1999) 464–472.

[5] T. Feder, P. Hell, S. Klein, and R. Motwani, List partitions, SIAM J. Comput. 16 (2003) 449–478.

[6] T. Feder, P. Hell, S. Klein, L. Tito Nogueira, and F. Protti, List matrix partitions of chordal graphs, Theoretical Computer Science 349 (2005) 52 – 66.

[7] T. Feder, P. Hell, and W. Xie, Matrix partitions with finitely many obstructions, manuscript 2006.

[8] D.P. Geoffrey and D. Sumner, The edge nucleus of a point-determining graph, J. Comb. Theory B 24 (1978) 189–201.

[9] P. Hell, ¿From graph colouring to constraint satisfaction: there and back again, Nesetril Birthday Book, Springer 2006.

[10] P. Hell and J. Nešetřil, On the complexity of H–colouring, J. Combin. Theory B 48 (1990) 92–110.

[11] P. Hell and J. Nešetřil, **Graphs and Homomorphisms**, Oxford University Press 2004.

[12] D. Sumner, Point determination in graphs, Discrete Math. 5 (1973) 179–187.

[13] D. Sumner, 1-factors of point determining graphs, J. Comb. Theory B 16 (1974) 35–41.

[14] D. Sumner, The nucleus of a point determining graph, Discr. Math. 14 (1976) 91–97.

[15] W. Xie, Obstructions to trigraph homomorphisms, M.Sc. Thesis, Simon Fraser University, 2006.