CRRAO Advanced Institute of Mathematics, Statistics and Computer Science (AIMSCS)

Research Report

Author (s): S. B. Rao and U. K. Sahoo

Title of the Report: **Graphs with specified Minimal Vertex Separator Graphs**

Research Report No.: RR2014-15

Date: May 28, 2014

Prof. C R Rao Road, University of Hyderabad Campus,
Gachibowli, Hyderabad-500046, INDIA.
www.crraoaimscs.org

Graphs with specified Minimal Vertex Separator Graphs

S. B. Rao[*] and U. K. Sahoo[†]
CR Rao Advanced Institute of Mathematics, Statistics and Computer Science,
Hyderabad, India - 500 046.

11:02, Wednesday 4^{th} June, 2014

Abstract

For two non-adjacent vertices x and y of a simple graph G, a xy vertex separator is a set of vertices $S \subseteq V(G)$, whose removal disconnects x and y. S is a minimal xy vertex separator if no proper subset of S is a xy vertex separator. This article characterizes some *good* classes of graphs, like the chordal graphs, based on the nature of the induced graph (MVS) on their minimal vertex separators which are C-free, where $C \in \mathcal{C}$ are small graphs on 2 and 3 vertices, namely, $\mathcal{C} = \{\overline{K_2}, K_2, \overline{K_3}, K_1 \cup K_2, P_3, K_3\}$. $\overline{K_2}$-free MVS graphs are the chordal graphs. Also $\overline{K_3}$-free, $K_1 \cup K_2$-free and P_3-free MVS graphs contain chordal graphs. We show that the forbidden graphs of these classes are the classic Truemper configurations, or their close relatives. We also study various graph characteristics like clique number, chromatic number, independence number, domination number, length of largest cycle and recognition of Hamiltonian cycle, some of which are polynomial in these classes. This is done using elimination orderings from Lex BFS and subdivisions.

Keywords: Minimum vertex separators, elimination ordering, Lex BFS, forbidden graph characterization.

1 Introduction

Let G be a simple graph with vertex set $V(G)$ and edge set $E(G)$. A *xy vertex separator* of non-adjacent vertices x and y is a set of vertices $S \subset V(G)$, whose removal disconnects them. S is a *minimal xy vertex separator* when no proper subset of it is a xy vertex separator, and let MVS be the graph induced by such a S on G i.e. MVS $= G[S]$. Throughout this article we assume G_{xy} is the MVS of G for non-adjacent vertices x and y. Also G_x and G_y are the components containing x and y respectively in $G \setminus G_{xy}$. By MVS is C-free, we mean that the MVS doesn't contain C as an induced subgraph. Let $\mathcal{C}$ be the set of forbidden MVS on 2 or 3 vertices, namely, $\mathcal{C} = \{\overline{K_2}, K_2, \overline{K_3}, K_1 \cup K_2, P_3, K_3\}$. We call a MVS G_{xy} is C-free if it doesn't contain C as an induced subgraph. We investigate

[*]Email: siddanib@yahoo.co.in
[†]Corresponding author. Email: umakant.iitkgp@gmail.com

the classes of graphs whose MVS are C-free for $C \in \mathcal{C}$. It should be noted that the property P of having MVS C-free is *hereditary*, i.e. every induced subgraph of any graph satisfying the property P also satisfies P.

Motivation: Chordal graphs are probably the most celebrated family of non-trivial perfect graphs and hence are widely investigated [1]. Thus many problems that are NP-Complete in other graphs, turn out to be polynomial, and usually linear, in chordal graphs. So, apart from the perfect graphs, our aim is to look into other such families of *good* graphs. It is well known, due to Dirac [2], that MVS of chordal graphs are cliques and hence are $\overline{K_2}$-free. In [3], Sadagopan proves that a graph is free of cycles with unique chord iff every MVS is an independent set. In this article we explore other classes of graphs based on such restrictions on MVS. We also refer to Garey and Johnson [4] for results on computational complexity, particularly NP Completeness.

A finite list of forbidden induced graphs usually yields a polynomial algorithm for testing membership in the class. Although for chordal graphs this list is infinite, namely cycles of length greater than 3, yet we have good recognition algorithms for it. But finding such characterization for class $C \in \mathcal{C}$ does not seem to be obvious. Let us illustrate the simple trick that we used on chordal graphs. Since MVS of chordal graphs are cliques, the MVS of chordal graphs are $\overline{K_2}$-free. The class of forbidden graphs are obtained by placing two vertices x and y on opposite sides of $\overline{K_2}$ and drawing paths of arbitrary length from x to y through their vertices. This gives us the class of forbidden induced subgraphs for chordal graphs. Minimality requires these paths to be disjoint except at x and y. So for chordal graphs our class of minimal forbidden graphs are the cycles C_n, $n \geq 4$. Our first step is to find similar results for graphs whose MVS are C-free, for $C \in \mathcal{C}$. In [5], Aboulker *et al.* found forbidden induced subgraphs for graphs where every induced subgraph has a vertex v whose neighbourhood $N(v)$ is $\mathcal{F}$-free, for a set of graphs $\mathcal{F}$. Not surprisingly the forbidden induced subgraphs we got, as well as in [5], are Truemper configurations or their close relatives. Truemper configurations play a key role in understanding the structures of perfect graphs [6].

One of the main reasons why analysing chordal graphs is *easy* is because of the presence of an *elimination ordering*. We call an ordering $(v_1, v_2, ..., v_n)$ of vertices an C-elimination ordering if for $i = 1, 2, ..., n$, every MVS in $G[\{v_1, v_2, ..., v_i\}]$ is C-free. As rightly pointed out in [5], sometimes designing efficient algorithms requires an elimination ordering along with a good structure in the neighbourhood, rather than a global description of the class. Our second step is to find such C-elimination ordering for $C \in \mathcal{C}$. Then using this, and few other transformations like *subdivisions* and proving NP-Completeness by *restriction* and *local replacement*, we find the various graph characteristics of class $C \in \mathcal{C}$. For any graph G, its i-subdivision G_i is formed by inserting i vertices (of degree 2) on every edge of G.

Another advantage of chordal graphs is that every such graph has a clique decomposition. Hence by iteratively adding a vertex joined to a clique, we can construct chordal graphs from a single vertex. These kinds of construction algorithms lead to fast algorithms for computations on graphs in this class. In [7], Trotignon and Vušković develop such construction techniques for graphs with no cycle with a unique chord. They give a structural definition for graphs with no cycles with a unique chord and present polynomial algorithms for recognition (of order $O(nm)$), finding clique number (of order $O(n+m)$) and chromatic number (of order $O(nm)$) for such graphs. They also prove that finding a maximal stable set for a graph in this class is NP-Complete.

In [5], Aboulker *et al.* provide a general method to prove the existence and compute efficiently elimination orderings in graphs using Lex BFS and a local decomposition property.

Results: First we prove a basic theorem on extending the MVS of an induced subgraph to the MVS of the whole graph. Then we show the connections between MVS and hereditary properties and minimal forbidden graphs. We also find minimal forbidden subgraphs for the classes of graphs whose MVS are C-free for $C \in \{\overline{K_2}, K_2, \overline{K_3}, K_1 \cup K_2, P_3, K_3\}$. Then we prove another basic result showing the structure of neighbourhood of graphs with C-free MVS. Using this result we prove that finding the clique number problems are in P for almost all $C \in \mathcal{C}$. Then with the help of subdivisions and restriction & local replacement in NP-Complete problems, we prove that other graph characteristics like finding the chromatic number, independence number, domination number, length of largest cycle and recognition of Hamiltonian cycle are NP-Complete for most of the cases, except ofcourse when they are polynomial. In Table 1, we summarize these graph characteristics, along with previous known results.

Organization: In Section 2, we state the fundamental theorem of this article: finding the minimal forbidden graphs for families of graphs with C-free MVS, for $C \in \mathcal{C}$. The proof is given in the Appendix. Then in Section 3, we present the Lex BFS and another fundamental result showing that the neighbourhood of each vertex in the elimination ordering is the join of a clique and a MVS of that class of graphs. Section 4 deals with analysis of various graph characteristics of such graphs with C-free MVS. Section 5 contains some examples and this article ends with some open questions in Section 6.

Table 1: Complexity of Graph Characteristics

MVS P-free	$\omega(G)$	$\chi(G)$	$\alpha(G)$	$\gamma(G)$	$l(G)$	Ham.
$\overline{K_2}$	P [1]	P [1]	P [1]	NPC [8]	NPC [t]	NPC [9]
K_2	P [7][t]	P [7]	NPC [7][t]	NPC [t]	NPC [t]	?
$\overline{K_3}$	NPC [t]	?	?	?	NPC [t]	NPC [t]
$K_1 \cup K_2$	P [t]	?	NPC [t]	NPC [t]	NPC [t]	NPC [t]
P_3	P [t]	?	NPC [t]	NPC [t]	NPC [t]	NPC [t]
K_3	P [t]	NPC [t]	NPC [t]	NPC [t]	NPC [t]	NPC [t]

$l(G)$ is the length of largest cycle and [t] represents this paper.

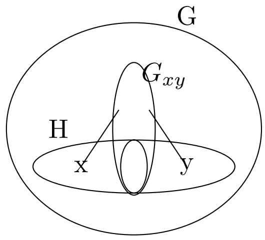


Figure 1

MVS and Hereditary Properties: Before looking into characterization of graphs on the basis of their MVS, we prove two important results relating the MVS of induced subgraphs to that of MVS of the graph. These results shall be repeatedly used in this article.

Theorem 1.1 *For an induced subgraph H of G, we can extend its MVS H_{xy} to a MVS G_{xy} of G.*

Proof: Refer Figure 1. Clearly $V(G) - V(H_x) \cup V(H_y)$ is a xy vertex separator (inducing H_{xy} as a subgraph) of G, so it also induces MVS of G. Also for every vertex $v_a \in V(H_{xy})$ there exists a xy path entirely contained in $H_x \cup H_y \cup \{v_a\}$ (else H_{xy} won't be minimal). So $v_a \in V(G_{xy})$. Hence H_{xy} is a induced subgraph of G_{xy}. $\square$

Using Theorem 1.1 we prove the following.

Corollary 1.2 *Given any hereditary property P, let $G(P)$ be the collection of finite graphs such that every G_{xy} has property P for every non-adjacent x and y, then every induced subgraph H of $G \in G(P)$ is also in $G(P)$ i.e. $G(P)$ is hereditary.*

Proof: Theorem 1.1 implies H_{xy} is an induced subgraph of G_{xy}. Since G_{xy} has hereditary property P for every non-adjacent x and y, H_{xy} also has the hereditary property P. So $H \in G(P)$. $\square$

So we have the following hereditary properties:

- MVS is $\overline{K_2}$-free, i.e. MVS induces a clique. This gives the class of chordal graphs.
- MVS is K_2- free, i.e. MVS induces an independent set. This gives the class of graphs with no cycle with a unique chord.
- MVS is $\overline{K_3}$-free. This gives the class of graphs with MVS G_{xy} such that the independence number $\alpha(G_{xy}) \leq 2$.
- MVS is $K_1 \cup K_2$-free, i.e. MVS induces a complete multipartite graph.
- MVS is P_3-free, i.e. MVS induces a collection of cliques.
- MVS is K_3-free, i.e. MVS is Δ-free.

From above hereditary properties, it is evident that the classes of graphs with $\overline{K_3}$-free MVS, $K_1 \cup K_2$-free MVS and P_3-free MVS contain the chordal graphs (i.e. graphs with $\overline{K_2}$-free MVS).

The following is a folklore result, due to Hemminger [10].

Lemma 1.3 *Given any hereditary property P, there is a set of minimal forbidden graphs.*

Truemper configurations: A *theta* is a graph consisting of two non-adjacent vertices a and b and three distinct ab-paths, no two of which intersect other than at a and b and any two ab paths induce a cycle. A *pyramid* is graph consisting of a triangle $\{b_1, b_2, b_3\}$, and a vertex $a \notin \{b_i\}$, for $i = 1, 2, 3$, such that a is connected to b_i by a path P_i, atmost one of them have length one, and any two such paths along with the connecting edge in triangle $\{b_1, b_2, b_3\}$ induces a cycle. A *prism* is a graph consisting of two disjoint triangles $\{a_i\}$ and $\{b_i\}$, for $i = 1, 2, 3$, connected by three distinct ab paths, and any two such paths along with the connecting edges in each of the triangles $\{a_i\}$ and $\{b_i\}$ induces a cycle. A *wheel* consists of an induced cycle C, called the rim, and a vertex x, called the center, that has atleast three neighbours in C. These four classes of graphs are known as *Truemper configurations* [11].

Chudnovsky and Seymour [12] prove that testing whether a graph has an induced theta is polynomial of order $O(n^{11})$. Extending this result, they also present a polynomial time algorithm, of order $O(n^{10})$, to detect whether a graph has an induced pyramid, which was already proved to be polynomial, of order $O(n^9)$, by Chudnovsky *et al.* [13]. Maffray and Trotignon [14] proved NP Completeness of detection of a prism as an induced subgraph. However, Chudnovsky and Kapadia [15] present a polynomial algorithm of order $O(n^{35})$ to detect whether a graph has an induced prism or theta.

Now we shall define some of the relatives of the truemper configurations that we shall come across. A 0-*wheel* is a wheel with atleast two non-consecutive spokes missing, i.e. a 3-set exists that induces a $\overline{K_3}$. A 1-*theta* is a theta with an extra edge joining interior vertices of two xy paths. A 1-*wheel* is a wheel with atleast one spoke missing , i.e. a 3-set exists that induces a $K_1 \cup K_2$, and the degree of the central vertex is atleast four. A 2-*theta* is a theta with an extra induced P_3 on interior vertex of each of the three xy paths. A 2-*pyramid* is a pyramid with an extra induced P_3 on interior vertex of each of the three xy paths. A 2-*prism* is a prism with an extra induced P_3 on interior vertex of each of the three xy paths. A *theta with* P_4 is a theta with an extra induced P_4 between four interior vertices of the three xy paths with two vertices (1^{st} and 4^{th}) in the central path and one each on the other paths. A 2-*wheel* is a wheel with the degree of the central vertex atleast four. A 3-*theta* is a theta with an extra induced K_3 on interior vertex of each of the three xy paths. A 3-*pyramid* is a pyramid with an extra induced K_3 on interior vertex of each of the three xy paths. A 3-*prism* is a prism with an extra induced K_3 on interior vertex of each of the three xy paths. A 1-*co-wheel* is two wheels joined at a common triangle with coinciding central vertices. A 2-*co-wheel* is two wheels joined at a common triangle with non-coinciding central vertices. These are shown in Figure 5 (*theta, pyramid, prism, 0-wheel*), Figure 7 (*1-theta, prism, 1-wheel*), Figure 9 (*2-theta, 2-pyramid, 2-prism, theta with* P_4*, 2-wheel*) and Figure 11 (*3-theta, 3-pyramid, 3-prism, 1-co-wheel, 2-co-wheel*).

2 Minimal Forbidden Graphs

Using the above mentioned configurations, we summarize our results on forbidden graph characterizations.

Theorem 2.1 *For a graph G and every MVS in it,*

- *If MVS is* $\overline{K_2}$*-free, then G is* C_n*-free for* $n \geq 4$*. (Also follows from Dirac [2]).*

- *If MVS is K_2-free, then G is C_n^1-free where C_n^1 is a cycle with unique chord. (Also follows from Sadagopan [3], also see Trotignon and Vušković [7]).*
- *If MVS is $\overline{K_3}$-free, then G is (theta, pyramid, prism, 0-wheel)-free.*
- *If MVS is $K_1 \cup K_2$-free, then G is (1-theta, prism, 1-wheel)-free.*
- *If MVS is P_3-free, then G is (2-theta, 2-pyramid, 2-prism, theta with P_4, 2-wheel)-free.*
- *If MVS is K_3-free, then G is (3-theta, 3-pyramid, 3-prism, 1-co-wheel, 2-co-wheel)-free.*

The proof of the above theorem is a bit lengthy, and hence, for sake of clarity, is given in the Appendix. Now we present some results on Lex BFS introduced by Rose, Tarjan and Lueker in [16]. However the algorithm given here is taken from the excellent reference book by Golumbic [1].

3 Lex BFS and Elimination Ordering

Algorithm
begin
1. assign the label $\emptyset$ to each vertex
2. **select**: pick an unnumbered vertex with largest label
3. $\sigma(i) \leftarrow i$ // Comment: v is assigned no. i
4. **update**: for each unnumbered vertex $w \in Adj(v)$ **do** add i to label(w)
end

What Lex BFS does is that it gives an ordering to vertices of the graphs by assigning them numbers from n to 1, based on the dictionary ordering of the labels assigned to its numbered neighbours. This gives an hierarchy layering among the vertices. The vertex who gets number n is in the top most layer 0. We shall call vertices by the number assigned to them. The next layer contains all the neighbours of n i.e. $n-1$ to $n-d$, d being the degree of n. The next layer contains all the neighbourhood of vertices of $n-1$ to $n-d$. Here also the order is maintained. The neighbourhood of $n-1$ gets higher numbers (or equal in case of common neighbours) than neighbours of other vertices in this layer, and so on.

One important observation is that every vertex in layer i is adjacent to atleast one vertex in its upper layer $i-1$. Apart from that its neighbourhood may contain vertices from layers i and $i+1$. However among the neighbourhood, the vertices in layer $i-1$ and some vertices that are already numbered in layer i determine the number assigned to that vertex.

In order to look for elimination ordering (as in chordal graphs) we need to look into the neighbourhood of last vertex. Hence, we are interested in the neighbourhood of vertex 1. Being the last vertex, it belongs to the final layer, and its neighbourhood are in the last layer and the layer above it. All its neighbours are numbered greater than 1, and clearly $N(1)$ is a vertex separator partitioning the graph into atleast two components; one is a singleton set containing 1 and the other component contains n. We shall now look into the other components.

If there are more than two components, the other components C_i must belong to the last layer. This is because the vertices in other layers belong to the component containing n, except ofcourse the neighbours of 1 in the penultimate layer that belong to the vertex separator.

Something more can be said about these components C_i. Since every vertex in C_i is connected to atleast one vertex in the penultimate layer, and the vertex separator disconnects it from the graph, it has to be adjacent only to a subset of $N(1)$ in the penultimate layer. Also since the vertices of C_i are numbered higher than 1, their neighbours in the penultimate layer should contain atleast one vertex with higher number that $N(1)$ in the penultimate layer or have exactly the same neighbourhood. Since the former is not possible (else C_i would be connected to n), all vertices of C_i are adjacent to $N(1)$ in the penultimate layer.

Now lets concentrate on $N(1)$. Atleast one of them belongs to the penultimate layer and rest belong to the last layer. Now let $N(1)^p$ represent the neighbourhood of 1 in the penultimate layer. Among the rest all vertices should be adjacent to atleast one vertex in the penultimate layer with higher number than $N(1)^p$ or be adjacent to every vertex in $N(1)^p$. Let $N(1)^+$ represent the neighbours of 1 in the last layer adjacent to atleast one vertex in the penultimate layer with higher number than any of $N(1)^p$. Let the rest neighbours of 1 be $N(1)^l$ which are all adjacent to each vertex in $N(1)^p$(else they wouldn't have been numbered greater than 1). So we have the following:

$$N(i) = N(1)^p + N(1)^+ + N(1)^l$$

Now we claim the following:

Lemma 3.1 $N(1)^p \cup N(1)^+$ *belong to the MVS.*

Proof: Since $N(1)$ is a vertex separator, it contains a MVS. We can't exclude any vertex of $N(1)^p$, else 1 and n get connected via that vertex. We have a similar argument for $N(1)^+$. Among vertices in $N(1)^l$, some may be adjacent to other higher numbered vertex in the last layer which is connected to n. Let such set of vertices be denoted by $N(1)^{l+}$, and the rest of $N(1)^l$ that are not connected to any vertex outside of $N(1)^l$ be denoted by $N(1)^{l-}$. Our MVS is exactly $N(1)^p \cup N(1)^+ \cup N(1)^{l+}$. □

Lemma 3.2 *$N(1)^{l-}$ induces a clique.*

Proof: Among $N(1)$, $N(1)^{l-}$ get the lowest numbers. Let the highest number that a vertex in $N(1)^{l-}$ get be i. Now it is easy to see that the next number got by a vertex in $N(1)^{l-}$ is $i-1$. Now this vertex has to be connected to i, else 1 would have got number $i-1$. Similarly vertex with number $i-2$ should be adjacent to i and $i-1$, and so on. Hence $N(1)^{l-}$ induces a clique. □

Now lets see the structure of $N(1)$, 1 is connected to two components, one is the MVS ($N(1)^p \cup N(1)^+ \cup N(1)^{l+}$), and the other is a clique ($N(1)^{l-}$). In order to have a more good structure in the neighbourhood of 1, we need the following result.

Lemma 3.3 *All the vertices of the clique are adjacent to all the vertices of MVS.*

Proof: As we have previously said that all vertices of $N(1)^l$, both $N(1)^{l+}$ and $N(1)^{l-}$, are adjacent to all vertices in $N(1)^p$, so every vertex of the clique is adjacent to all vertices of $N(1)^p$. Now look at vertices of $N(1)^+$, each of them has to be adjacent to every vertex in $N(1)^{l-}$, else 1 would have got a higher number. Similarly for $N(1)^{l+}$, the same argument holds. So every vertex of the clique is adjacent to every vertex of MVS. □

Lemma 3.4 *The clique can be recognised in polynomial time.*

Proof: Arrange $N(1)$ in descending order of the Lex BFS. Now look into neighbours of each vertex in $N(1)$. Those vertices whose every neighbour is in $N(1)$ form the clique. This is because each of $N(1)^p$, $N(1)^+$ and $N(1)^{l+}$ have a neighbour out of $N(1)$, only $N(1)^{l-}$ (which is the clique) hasn't. □

Using the Lex BFS ordering, we use the structure of neighbourhood of vertices, as got in the above Lemmas 3.1, 3.2 & 3.3 to get the following theorem.

Theorem 3.5 *If a graph G has C-free MVS, then it has a Lex BFS ordering from n to 1 satisfying, every neighbourhood of vertex i in $G \setminus G[\{1, 2, ..., i-1\}]$, $N(i) = N(i)_1 + N(i)_2$, where $N(i)_1$ is a clique and $N(i)_2$ is the MVS that is C-free, and the operator $+$ denotes the usual graph join operation.*

Given such an ordering it is easy to check whether the neighbourhood contains a clique, and is adjacent to every vertex of MVS. Also to recognize any of these classes of graphs, we need to determine whether the MVS of the structure is C-free. Below are the summary of the structures of MVS of the six cases we discussed above.

- MVS is $\overline{K_2}$-free, i.e. MVS induces a clique. This can be checked in polynomial time.
- MVS is K_2-free, i.e. MVS induces an independent set. This also can be checked in polynomial time.
- MVS is $\overline{K_3}$-free, i.e. for MVS $\alpha(G_{xy}) \leq 2$. So its complement is Δ-free. This also can be checked in polynomial time.
- MVS is $K_1 \cup K_2$-free, i.e. MVS induces a complete multipartite graph. This also can be checked in polynomial time.
- MVS is P_3-free, i.e. MVS induces a collection of cliques. This also can be checked in polynomial time.
- MVS is K_3-free. This also can be checked in polynomial time.

For the classic case of chordal graphs i.e. $\overline{K_2}$-free MVS, a Lex BFS will give a sequence from n to 1. As proven above, the neighbourhood should contain a MVS and a clique, which is adjacent to every vertex of the MVS. However in this case the MVS is the clique. So in order to check if a graph is chordal or not, it suffices to check whether the neighbourhood induces a clique.

For graphs with MVS K_2-free, Trotignon and Vušković [7] give a polynomial recognition algorithm of order $O(nm)$. Here we shall use our Lex BFS ordering to give an alternate polynomial recognition algorithm. We have a sequence from n to 1. As proven above, the neighbourhood should contain a MVS and a clique, which is adjacent to every vertex of the MVS. In this case the MVS is an independent set. So the vertices of the clique will have the maximum degree. Now for the remaining vertices in the neighbourhood, check whether all of them are adjacent to the clique and are independent. This can be done in a polynomial time.

Now lets look at the graphs with $K_1 \cup K_2$-free MVS i.e. complete multipartite graphs. After doing a Lex BFS on such a graph we shall get a sequence from n to 1. Start with the vertex 1 and look at its neighbours which is composed of a MVS and a clique adjacent to all vertices of the MVS. Please note that the MVS is also a complete t-partite graph. It is also easy to see that the clique, say on m vertices and the MVS form a complete $(t+m)$-partite graph (the clique can be viewed as a complete m-partite graph). Now remove vertex 1. The rest graph is also a multipartite graph and we go on repeating this process. So we have a sequence of vertices whose neighbourhood is a complete multipartite graph. In order to check whether a graph has $K_1 \cup K_2$-free MVS, we look at the neighbourhood of the last vertex in the Lex BFS. Take any vertex in this neighbourhood and look at the non adjacent vertices to this vertex. They should be independent. Now take another vertex in the neighbourhood of 1 which doesn't belong to this partition, and follow these steps. After #partition steps we shall know whether the neighbours induce a complete multipartite graph. Repeating this for all the vertices in the ordering will tell us whether the graph has $K_1 \cup K_2$-free MVS. This can be done in a polynomial time.

Similarly for graphs with $\overline{K_3}$-free MVS i.e. whose $\alpha(G_{xy}) \leq 2$, we can do a Lex BFS. The neighbourhood of the vertex 1 is a MVS and a clique, adjacent to every vertex of the MVS. We take the complement of the graph induced by neighbourhood of 1 and check whether it is Δ-free. Go on doing this for other vertices in the ordering. This will tell us whether the graph has $\overline{K_3}$-free MVS, and can be done in polynomial time.

For graphs with P_3-free MVS, the MVS induce a collection of cliques. So the vertices of the clique (due to Theorem 3.5) will have the maximum degree. Now for the remaining vertices in the neighbourhood, check whether all of them are adjacent to the clique. If so pick a vertex and collect all adjacent vertices not in the clique. Check whether this induces a clique. Go on doing this till all vertices are exhausted. This can be done in a polynomial time.

Finally for graphs with K_3-free MVS, just check whether neighbourhood of each vertex in the ordering after recognising the clique is K_3-free.

However this doesn't help in the recognition of the above classes, except chordal graphs, due to the following arguments.

Considering a generalized converse statement of the Theorem 3.5, we have the following definition.

Definition: A graph G is said to have the property C^*, if it has an ordering such that every vertex i in the ordering has a neighbourhood, in $G \setminus G[\{1, 2, ..., i-1\}]$, which is a graph join between a clique and a MVS that is C-free. Such an ordering is called as C^* *sequence.*

The C^* *sequence* is a variant of an ordering defined by Aboulker *et al.* [5] known as C *elimination ordering* (i.e. ordering in $G[\{v_1, v_2, ..., v_i\}]$ where neighbourhood is C-free). They also have introduced a stronger concept known as *locally* C *decomposable* and have completely characterized graphs for $C \in \{\overline{K_3}, K_1 \cup K_2, P_3\}$. When C contains complete graphs, then a complete graph might not have C elimination ordering. Hence they don't consider C to be a complete graph. However, since we are looking into MVS, we consider all C on 2 and 3 vertices.

For class $C = \overline{K_2}$, our forbidden graphs matches with that in [5]. For classes $C \in \{\overline{K_3}, K_1 \cup K_2\}$, it can be easily checked that the forbidden graphs in [5], are subsets of our forbidden graphs. So for $C \in \{\overline{K_2}, \overline{K_3}, K_1 \cup K_2\}$ our classes of graphs is a subset of the classes in [5]. So their C *elimination ordering* can govern our classes of graphs. However for $C = P_3$, the graph K_4 with only edge replaced by a P_3 is a forbidden graph for the corresponding class defined in [5], but its MVS is P_3 free, hence not a forbidden graph for us. Hence the C *elimination ordering* defined in [5] cannot govern graphs with P_3 free MVS. Hence we need our elimination ordering.

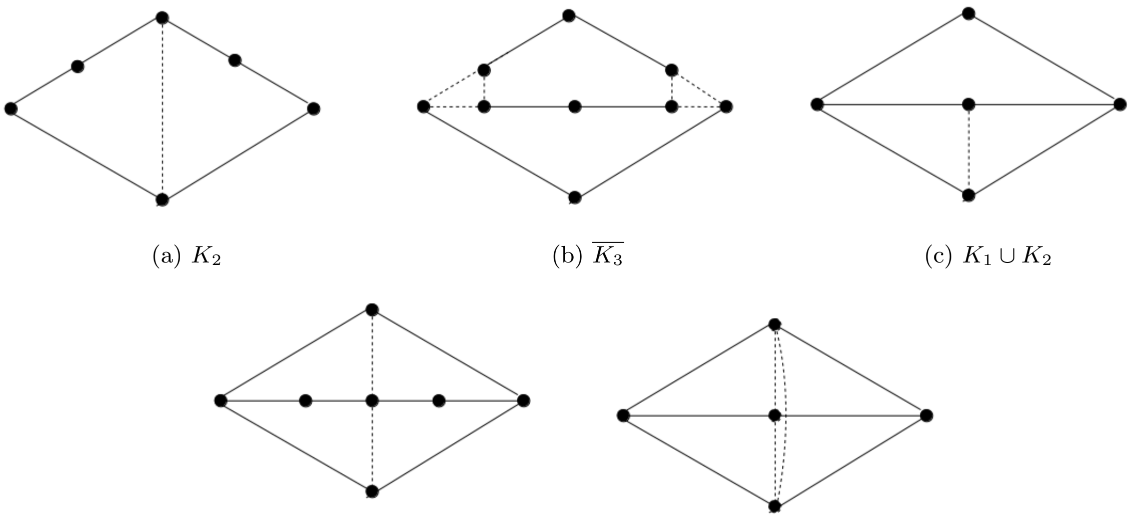
Clearly the classes of graphs that have C-free MVS belong to this class satisfying C^* property, by Theorem 3.5. But the following lemma highlights the drawback of such an approach.

Lemma 3.6 *Even if every ordering of vertices gives the* C^* *sequence, then the original graph need not have* C*-free MVS for* $C \in \{K_2, \overline{K_3}, K_1 \cup K_2, P_3, K_3\}$.

Proof: We prove the above statement by explicitly giving examples of family of graphs for each case where every any ordering (particularly Lex BFS) gives the C^* sequence, although the MVS of the graph contains C. Figure 2 contains graphs which contain C in their MVS ($C \in \{K_2, \overline{K_3}, K_1 \cup K_2, P_3, K_3\}$), but every ordering (particularly Lex BFS) is a C^* sequence. For each graph a family of such graphs can be obtained by replacing the bold lines by paths of arbitrary length. $\square$

In particular, we have the following Corollary and remark.

Corollary 3.7 *Even if every Lex BFS gives the* C^* *sequence, the original graph need not have* C*-free MVS for* $C \in \{K_2, \overline{K_3}, K_1 \cup K_2, P_3, K_3\}$.



(a) K_2 (b) $\overline{K_3}$ (c) $K_1 \cup K_2$

(d) P_3 (e) K_3

Figure 2: C^* sequence not enough for recognition.

Remark: Having such an ordering is not enough for the recognition of the classes of graphs we consider, except the chordal graphs, however we shall use this ordering in order to find the clique numbers of some cases in polynomial time. In case of the chordal graphs we suspect that the symmetry in its minimal forbidden graphs is the reason that it escapes Lemma 3.6.

4 Graph Characteristics

4.1 Independence Number

Now we shall look into finding the independence number of graphs belonging to the classes discussed above. It is well known that independence number of chordal graphs can be found in linear time [1]. However, the following result proves NP-Completeness for many other cases. And this particular technique is used repeatedly in this article.

Lemma 4.1 *For a hereditary property P, suppose each minimal forbidden graph has two adjacent vertices of degree ≥ 3, then the determination of independence number $\alpha(G)$ is NP-Complete for graphs with property P.*

Proof: Let G be a graph and P be the hereditary property. Let $I(G)$ be its independent set. We consider the subdivision G_2 of G. For every edge (P_2) in G, we have a P_4 in G_2. In G_2 there aren't any two adjacent vertices of degree ≥ 3 and therefore it has property P. Let $I(G_2)$ be the independent set of G_2. We claim that $\alpha(G_2) = m_G + \alpha(G)$, where m_G is the number of edges in G. Each P_4 must have atleast 1 and atmost 2 independent vertices. If a P_4 has just 1 vertex in $I(G_2)$ then one of its end vertices (farther from its independent vertex) should be adjacent to a vertex in $I(G_2)$. Our aim is to find the maximum cardinality of an independent set. So we shall try to maximize those P_4's which have 2 vertices in $I(G_2)$. We know that for a maximum independent set the number of edges originating from them is maximum. So $\#P_4$'s originating from $I(G)$ in G_2 is also maximum. We take two vertices from these paths in the following manner. Take all $I(G)$ in $I(G_2)$. For each P_4 originating from $I(G)$, take the vertex at a distance 2 from $I(G)$. For the rest P_4's take any interior vertex. So we took all vertices in $I(G)$ along with a vertex from each P_4. So $\alpha(G_2) = m_G + \alpha(G)$. Since the family of graphs G_2 is contained in family of graphs with property P, by local replacement, the determination of independence number $\alpha(G)$ is NP-Complete for graphs with property P. $\square$

So we have the following corollary.

Corollary 4.2 *The determination of independence number $\alpha(G)$ is NP-Complete for graphs with K_2-free MVS, $K_1 \cup K_2$-free MVS, P_3-free MVS and K_3-free MVS.*

These results are reflected in Table 1.

4.2 Domination Number

For the class of chordal graphs it is known that the domination number problem is NP-Complete, due to Booth and Johnson [8]. The following result proves NP-Completeness for many other cases.

Lemma 4.3 *For a hereditary property P, suppose each minimal forbidden graph has two adjacent vertices of degree ≥ 3, then the determination of domination number $\gamma(G)$ is NP-Complete for graphs with property P.*

Proof: Let G be a graph and P be the hereditary property. Let $S(G)$ be its minimum dominating set. We consider the subdivision G_3 of G. For every edge (P_2) in G, we have a P_5 in G_3. In G_3 there aren't any two adjacent vertices of degree ≥ 3 and therefore it has property P. Let $S(G_3)$ be the minimum dominating set of G_3. We claim that $\gamma(G_3) = m_G + \gamma(G)$, where m_G is the number of edges in G. In G_3 we have three kinds of P_5's; the P_5's between vertices of $S(G)$, the P_5's between vertices of $V(G) - S(G)$ and the P_5's between a vertex of $S(G)$ and a vertex of $V(G) - S(G)$. If the end vertices of a P_5 is dominated by external vertices then the central vertex dominates the other vertices. If none or only one vertex of P_5 is dominated by external vertices, then we need two dominating vertices. So each P_5 contributes atleast one to $\gamma(G_3)$. Apart from one vertex per P_5 we need to take some extra vertices in the dominating set. Our aim is to minimize this extra vertices. We make use of the fact that the minimum dominating set of G contains the minimum number of vertices that dominate all of G.

Amongst the P_5's between vertices of $V(G)-S(G)$, the central vertex belongs to the minimum dominating set. If central vertex doesn't belong to the dominating set, then the dominating set will contain two vertices in the P_5. Now the extreme vertices of these P_5's have to be dominated. So the fourth vertex in the P_5's between a vertex of $S(G)$ and a vertex of $V(G) - S(G)$ i.e. the vertex at a distance 3 from $S(G)$, will belong to the minimum dominating set. Now $S(G)$ and its adjacent vertices in these P_5's are to be dominated. So we take $S(G)$ in $S(G_3)$. Now for the left P_5's between vertices of $S(G)$, any of the internal vertices per P_5 can belong to $S(G_3)$. So $S(G_3)$ contains $S(G)$ and one vertex per P_5. Hence, $\gamma(G_3) = m_G + \gamma(G)$. Since the family of graphs G_3 is contained in family of graphs with property P, by local replacement, the determination of domination number $\gamma(G)$ is NP-Complete for graphs with property P. $\square$

So we have the following corollary.

Corollary 4.4 *The determination of domination number $\gamma(G)$ is NP-Complete for graphs with K_2-free MVS, $K_1 \cup K_2$-free MVS, P_3-free MVS and K_3-free MVS.*

It should be noted that G_3 is a bipartite graph, hence the following results.

Lemma 4.5 *For a hereditary property P, suppose each minimal forbidden graph has two adjacent vertices of degree ≥ 3, then the determination of domination number $\gamma(G)$ is NP-Complete for bipartite graphs with property P.*

Corollary 4.6 *The determination of domination number $\gamma(G)$ is NP-Complete for bipartite graphs with K_2-free MVS, $K_1 \cup K_2$-free MVS, P_3-free MVS and K_3-free MVS.*

These results are reflected in Table 1.

4.3 Length of Largest Cycle

The following result proves NP-Completeness of the largest cycle problem for many cases.

Lemma 4.7 *For a hereditary property P, suppose each minimal forbidden graph has two adjacent vertices of degree ≥ 3, then the determination of length of longest cycle $l(G)$ is NP-Complete for graphs with property P.*

Proof: Let G be a graph and P be the hereditary property. Let $l(G)$ be the length of its largest cycle. We consider the subdivision G_1 of G. For every edge (P_2) in G, we have a P_3 in G_1. In G_1 there aren't any two adjacent vertices of degree ≥ 3 and therefore it has property P. Let $l(G_1)$ be the length of its largest cycle of G_3. We claim that $l(G_1) = 2l(G)$. Let the largest cycle in G be $v_1v_2...v_{l(G)}v_1$. Then the largest cycle in G_1 will be $v_1u_{12}v_2u_{23}...v_{l(G)}u_{l(G)1}v_1$, where u_{ij} is the new vertex introduced in G_1 between vertices v_i and v_j of G. So its length is $2l(G)$. Since the family of graphs G_1 is contained in family of graphs with property P, by local replacement, the determination of length of longest cycle $l(G)$ is NP-Complete for graphs with property P. $\square$

Later in Corollary 5.2, we shall see that the Hamiltonian cycle problem is NP-Complete for graphs with $\overline{K_2}$-free and $\overline{K_3}$-free MVS, so again by restriction, the problem of the largest cycle is NP-Complete.

In lieu of the above results we have the following Corollary.

Corollary 4.8 *The determination of length of largest cycle $\gamma(G)$ is NP-Complete for graphs with $\overline{K_2}$-free, K_2-free MVS, $\overline{K_3}$-free, $K_1 \cup K_2$-free MVS, P_3-free MVS and K_3-free MVS.*

It should also be noted that G_1 is a bipartite graph, hence the following results.

Lemma 4.9 *For a hereditary property P, suppose each minimal forbidden graph has two adjacent vertices of degree ≥ 3, then the determination of length of longest cycle $l(G)$ is NP-Complete for bipartite graphs with property P.*

Corollary 4.10 *The determination of length of largest cycle $\gamma(G)$ is NP-Complete for bipartite graphs with K_2-free MVS, $K_1 \cup K_2$-free MVS, P_3-free MVS and K_3-free MVS.*

These results are reflected in Table 1.

4.4 Clique Number

4.4.1 Clique number of graphs with $\overline{K_3}$-free MVS

The clique number of any graph is the independence number of its complement. It is known that the problem of finding independence number of Δ-free graphs is NP-Complete (due to Poljak [17] and restriction). So the problem of finding clique number of $\overline{K_3}$-free graphs in also NP-Complete. But what we are interested in is finding the clique number of graphs with $\overline{K_3}$-free MVS. But graphs with $\overline{K_3}$-free MVS already contain $\overline{K_3}$-free graphs. Hence, by restriction, clique number problem is NP-Complete for graphs with $\overline{K_3}$-free MVS. We also give an alternative proof.

Lemma 4.11 *The clique number problem is NP-Complete even for graphs with $\overline{K_3}$-free MVS.*

Proof: Given any $\overline{K_3}$-free graph H, we shall construct a graph G containing H as a MVS, and whose other MVS are also $\overline{K_3}$-free. Construction: Take two vertices x and y, and join them to every vertex of H. So the MVS for x and y is H which is $\overline{K_3}$-free. For any two non-adjacent vertices in H, the MVS will contain x and y along with some more vertices in H. Since x and y are adjacent to all such vertices, and H is $\overline{K_3}$-free, no three vertices will induce $\overline{K_3}$. Hence we have a polynomial transformation from a $\overline{K_3}$-free graph to a graph with $\overline{K_3}$-free MVS. Hence this problem is also NP-Complete. $\square$

4.4.2 Clique number of graphs with $K_1 \cup K_2$-free MVS

We already know that the complete t-partite graphs belong to this class. We have the following lemma.

Lemma 4.12 *The clique number problem is polynomial for the class of graphs with $K_1 \cup K_2$-free MVS.*

Proof: We take a Lex BFS of the graph giving us a ordering from 1 to n. We then take a maximal clique and take the vertex that first appears in the Lex BFS. Remove all the vertices before that. No vertices of the maximal clique are removed since they occur later in the ordering. So this clique is still present in the neighbourhood of that vertex. Since in a complete t-partite graph all the cliques are of size t, we can compute the clique sizes of neighbourhood of each vertex in the Lex BFS ordering. The maximum of these give the clique size.

4.4.3 Clique Number of graphs with K_3-free MVS

This section deals with the following theorem.

Theorem 4.13 *Clique number $\omega(G)$ of a graph with K_3-free MVS can be found in polynomial time.*

We need the following lemma to prove the above theorem.

Lemma 4.14 *In a graph with K_3-free MVS, given any Δ, there is a unique maximal clique containing this Δ.*

Proof: Suppose there are atleast two maximal cliques containing this Δ and let G_1 and G_2 be the component of these cliques excluding the Δ. It is easy to see that for $x \in G_1$ and $y \in G_2$, the MVS of these two vertices contains the Δ. Hence we have a contradiction. So there is a unique maximal clique containing this Δ. □

Any three vertices in a clique induces a Δ. So we have the following corollary.

Corollary 4.15 *In a graph with K_3-free MVS, no two maximal cliques have three vertices in common.*

Proof of Theorem 4.13 We present an explicit algorithm to find the clique number $\omega(G)$ of a graph with K_3-free MVS.

Algorithm:
1. $S = \emptyset$, $E = \emptyset$
2. for $\Delta x_i x_j x_k$ in G where $x_i, x_j, x_k \notin E$
3. $\quad S = S \cup \{x_i, x_j, x_k\}$
4. $\quad$ while x_l is adjacent to any three elements of S
5. $\quad\quad S = S \cup \{x_l\}$
6. $\quad CLIQUE = CLIQUE \cup \{S\}$
7. $\quad CLIQUENUMBER = CLIQUENUMBER \cup \{|S|\}$
8. $\quad E = E \cup S$

9. $S = \emptyset$
10. Return maximum($CLIQUENUMBER$)

S contains the vertices of each maximal clique in the graph. E is used to collects all the vertices contained in S in any of the previous iterations so as to ensure that a maximal clique is got only once, hence making our algorithm more efficient. In line 2, we take triangle $x_i x_j x_k$ so that they don't belong to E, else we would get the same maximal clique. In line 3, we add these vertices to S. In lines 4 and 5, we go on adding those vertices x_l to S, which are adjacent to any three vertices in S. We go on doing this till no such x_l exists. S has a Δ, and after that we add vertices that are adjacent to three vertices of S. Due to Lemma 4.14, all vertices in S form a maximal clique, particularly the unique maximal clique containing the initial Δ. In lines 6 and 7, we add this maximal clique and its clique size in $CLIQUE$ (a set of sets containing the vertices of each maximal cliques) and $CLIQUENUMBER$ (a set containing the clique sizes) respectively. In line 8, we add the vertices of S to E so as to ensure that we don't consider any triangle in this maximal clique in further iterations which ensures that we don't get the same maximal clique once again. In line 9, we initialize S back to $\emptyset$ for finding the next maximal clique in the graph. After considering all such Δ's in the graph, we have found out all maximal cliques and their respective clique sizes. We output the maximum of all clique sizes in line 10. So we have the clique number of the graph. Hence finding the clique number $\omega(G)$ of such graphs with Δ-free MVS is polynomial. $\square$

These results are reflected in Table 1.

Now, we construct graphs with K_3-free MVS having arbitrary clique number. The following lemma gives the construction.

Lemma 4.16 *A graph with K_3-free MVS can have arbitrary clique number $\omega(G)$.*

Proof: We take a K_3-free graph and connect it by edge to a arbitrary clique of size n. So our graph has a clique component and a Δ-free component. Every MVS is found *wrt* two non-adjacent vertices x and y. Both of them cannot belong to the clique component, since they will be adjacent. If both belong to Δ-free component, their MVS is already Δ-free. Now *wlog* let x belong to the clique component and y to Δ-free component. Two cases arise: x is adjacent to the Δ-free component or it is not. In the former case MVS is just a K_1, particularly the vertex of the Δ-free component connected to the K_n; whereas in the latter, $K_n - x$ will be a vertex separator, and again MVS will be K_1, particularly the vertex of K_n connected to the Δ-free component. The other vertex separators will lie in the Δ-free component, hence MVS will also be Δ-free. So for every x and y, their MVS G_{xy} is K_3-free, and its clique number is arbitrary. $\square$

4.5 Chromatic Number

4.5.1 Chromatic Number of graphs with Δ-free MVS

As discussed previously, it is well known, due to Král *et al.* [18], that the chromatic number problem is NP-Complete for Δ-free graphs. Since Δ-free graphs is a subset of class of graphs with K_3-free MVS, by restriction, we have the following result.

Lemma 4.17 *The chromatic number problem is NP-Complete for graphs with K_3-free MVS.*

4.5.2 On the structure of G when G_{xy} is Δ-free

From Lemma 4.14, we have no two maximal cliques have a Δ in common i.e. no two maximal cliques have three vertices in common. So any two intersecting maximal cliques will have K_1 or K_2 in common. Let G_1, G_2, ..., G_n be the corresponding components of maximal cliques got after removing all the common K_1's and K_2's, so they too are cliques. Clearly the whole graph G can be covered by its vertices. We now group the vertices according to the maximal cliques they form. So G can be covered by the set of maximal cliques, including K_1's and K_2's. Please note that these K_1's and K_2's are also maximal cliques. Hence G can be represented by available connections between G_i's by K_1 or K_2.

These results are reflected in Table 1.

5 Examples of graphs with C-free MVS

It is known that chordal graphs have MVS $\overline{K_2}$-free. Infact since the MVS of chordal graphs are cliques, their MVS also are $\overline{K_3}$-free, $K_1 \cup K_2$-free and P_3-free. The class of graphs with K_2-free MVS are well characterized by Trotignon and Vušković in [7]. Lets call these graphs Trotignon graphs. It is also easy to see that the join of such graphs and any independent set will give K_3-free MVS. So now we have a good number of examples for each case of graphs with MVS C-free. Not only that, by restriction, these help us prove NP-Completeness of other characteristics in these classes. Since chordal graphs are contained in graphs whose MVS are $\overline{K_3}$-free, $K_1 \cup K_2$-free and P_3-free, by restriction, we have the following theorem.

Theorem 5.1 *Any graph characteristic that is NP-Complete for chordal graphs is NP-Complete for graphs whose MVS are $\overline{K_3}$-free, $K_1 \cup K_2$-free and P_3-free.*

5.1 Graph Characteristics continued: Hamiltonian Cycle

Since Hamiltonian cycle problem is NP-Complete for chordal graphs, due to Colbourn and Stewart [9], we have the following theorem.

Corollary 5.2 *The Hamiltonian cycle problem is NP-Complete for graphs with $\overline{K_2}$-free MVS, $\overline{K_3}$-free MVS, $K_1 \cup K_2$-free MVS and P_3-free MVS.*

Using this trick we can alternately prove that the determination of domination number $\gamma(G)$ is NP-Complete for these cases, yet we have given the alternate proof for sake of completeness.

Due to Krisnamoorty [19], it is known that Hamiltonian cycle problem is NP-Complete for bipartite graphs, which are Δ-free. Since Δ-free graphs is a subset of class of graphs with K_3-free MVS, by restriction, we have the following result.

Lemma 5.3 *The Hamiltonian cycle problem is NP-Complete for graphs with K_3-free MVS.*

These results are reflected in Table 1.

Now drawing a parallel with Aboulker *et al.* [5], since the classes for $C \in \{\overline{K_2}, \overline{K_3}, K_1 \cup K_2\}$ defined here are a subset of graphs of their classes, their polynomial algorithms for graph characteristics, namely clique number, also applies to ours, whereas for graph characteristics which are NP Complete in our class, is by restriction, also NP-Complete for their classes. These include the following: the determination of independence number and the dominating number is NP-Complete for $C = K_1 \cup K_2$, determination of length of largest cycle and the Hamiltonian problem is NP-Complete for $C \in \{\overline{K_2}, \overline{K_3}, K_1 \cup K_2\}$.

6 Open Problems

The recognition problem of whether a graph has MVS C-free for $C \in \{\overline{K_3}, K_1 \cup K_2, P_3, K_3\}$ is not known. One of the approaches to find structural characterization of such classes of graphs using the techniques of Trotignon and Vušković in [7]. Another alternative is to recognize the forbidden subgraphs present in it. There exists polynomial algorithms for some of them like theta, pyramid and theta or prism by Chudnovsky and others [12, 13, 15]. Although it is NP-Complete to recognise some of the structures, yet since we need to recognise any one of a set of forbidden minimal graphs, it might make things easier (as in the recognition of theta or prism is polynomial, although recognising prism is NP-Complete [15]). However detection algorithms aren't known for the other kinds of configurations we saw in Theorem 2.1. Although seems a lot involved, this is a possible direction for future work. Another, relatively easier, set of problems would be to resolve the unknown graph characteristics in Table 1, particularly the determination of chromatic number for the class of graphs with MVS C-free for $C \in \{\overline{K_3}, K_1 \cup K_2, P_3\}$, although we suspect it to be NP-Complete. Refer Table 1 for the other unknown graph characteristics.

7 Acknowledgement

The authors would like to thank DST CMS GoI Project No. SR/S4/MS: 516/07 dated 21.04.2008 and its extension for their research support.

References

[1] M. C. Golumbic, Algorithmic Graph Theory and Perfect Graph, 2^{nd} Ed., Elsevier 2004.

[2] G. A. Dirac, On rigid circuit graphs. Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg, **25** (1961), 71-76.

[3] N. Sadagopan, Impact of Graph Operations on Connectivity, Ph.D. Thesis, Indian Institute of Technology Madras, 2011.

[4] M. R. Garey and D. S. Johnson, Computers and Intractability: A guide to the theory of NP Completeness, W. H. Freeman and Company (1979).

[5] P. Aboulker, P. Charbit, N. Trotignon and K. Vušković, Vertex elimination orderings for hereditary graph classes, arXiv:1205.2535 [cs.DM], 2013.

[6] K. Vušković, The world of hereditary graph classes viewed through Truemper configurations, Surveys in Combinatorics, London Mathematical Society Lecture Note Series 409 , Ed. S.R. Blackburn, S. Gerke and M. Wildon, Cambridge University Press, pages 265-325, 2013.

[7] N. Trotignon and K. Vušković, A Structure Theorem for Graphs with No Cycle with a Unique Chord and Its Consequences, Journal of Graph Theory, **63(1)** (2010), 31-67.

[8] K.S. Booth and J.H. Johnson, Dominating sets in chordal graphs, SIAM Journal of Computing, **11** (1982), 91-199.

[9] C.J. Colbourn and L.K. Stewart, Dominating cycles in series-parallel graphs, Ars Combinatoria, **1pA** (1985), 107-112.

[10] D. L. Greenwell, R. L. Hemminger and J.Kleitman, Forbidden Subgraphs, Proceedings of 4^{th} South East Conference of Graph Theory and Computing, Florida Atlantic University (1973), 389-394. MR 0354446

[11] K. Truemper, Alpha-balanced graphs and matrices and GF(3)-representability of matroids, *Journal of Combinatorial Theory Series B* **32** (1982), 112-139.

[12] M. Chudnovsky and P. Seymour, The three in a tree problem, *Combinatorica* **30** (2010), 387-417.

[13] M. Chudnovsky, G. Cornuéjols, X. Liu, P. Seymour, and K. Vušković, Recognizing Berge Graphs, *Combinatorica* **25** (2005), 143-187.

[14] F. Maffray and N. Trotignon, Algorithms for perfectly contractile graphs, *SIAM Journal on Discrete Math.*, **19** (2005), 553-574.

[15] M. Chudnovsky and R. Kapadia, Detecting a theta or a prism, *SIAM Journal on Discrete Math.*, **22** (2008), 1164-1186.

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

[17] S. Poljak, A note on the stable sets and coloring of graphs, Commentationes Mathematicae Universitatis Carolinae **15** (1974), 307-309.

[18] D. Král, J. Kratochvíl, Z. Tuza and G. J. Woeginger, Complexity of Coloring Graphs without Forbidden Induced Subgraphs, Lecture Notes in Computer Science **2204** (2001), 254-262.

[19] M.S. Krisnamoorty, An NP-hard problem in bipartite graphs, SIGAT News **7** (1976), 26-26.

A Appendix: Minimal Forbidden Graphs [Proof of Theorem 1]

A.0.1 Minimality Transformation

Before starting to find out the minimal forbidden subgraphs, we shall give a condition called as the *minimality transformation*. This technique is used to prove that a given graph is not minimal in terms of forbidden MVS by showing that by appropriately taking x and y, usually by shifting them to the highlighted vertices, and then deleting some vertices within the dotted circle, to get a further smaller graph. Hence, such graphs violate the *minimality conditions*. We follow the following conventions: the highlighted (bigger) vertex is the new position of x, if two vertices are highlighted by same color one is x and other is y; the vertices inside the dotted circle can be deleted; and the new non-trivial MVS is highlighted by stars.

Now we shall look into minimal forbidden graph characterization.

A.1 Forbidden Graph Characterization

We consider characterization of some particular class of graphs based on some small forbidden graphs, which we shall prove whenever it isn't obvious. Then we construct minimal forbidden graphs of such graphs whose MVS are those particular classes. Also if a class of graphs has MVS as either one of the following cases then its forbidden graph characterization can be found by collecting all the forbidden graphs of each individual class.

Proof of Theorem 2.1:

A.1.1 $\overline{K_2}$ free

A complete graph doesn't contain $\overline{K_2}$. So a graph with MVS $G_{xy} = K_n$ has $\overline{K_2}$-free MVS. So a minimal forbidden graph will be C_n, for $n \geq 4$. This is obtained by placing x and y on opposite sides of $\overline{K_2}$ and drawing paths of arbitrary length from x to y. This is shown in Figure 3.1. Such classes of graphs are well known as the chordal graphs. Please note that a small dash over the edge represents a path of arbitrary length (≥ 1). For more details on this characterization please refer [1].

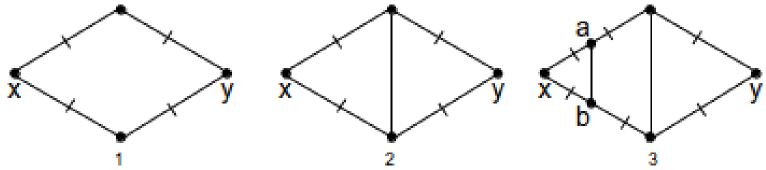


Figure 3: MVS with 2 vertices

A.1.2 K_2 free

Similarly an independent set is K_2 free. So a graph with MVS $G_{xy} = \overline{K_n}$ has K_2-free MVS. So a minimal forbidden graph will be as shown in Figure 3.2, which is obtained, similarly, by placing x and y on opposite sides of K_2 and drawing paths of arbitrary length from x to y. In fact, this cycle with unique chord, is the only minimal forbidden graph for $G_{xy} = \overline{K_n}$. For more details on this characterization please refer [7].

Lemma A.1 *Cycles with unique chord K_2 are the only minimal forbidden graphs for graphs with MVS as independent sets.*

Proof: Suppose the chord isn't unique, then there exists atleast another chord ab. This will lie in one side of K_2, because if it crosses, K_2 ceases to be MVS. Since ab lies on one side of K_2, then we can shift x to a and delete x (as shown in Figure 1.3); so it ain't minimal. Hence we need an unique chord.

A.1.3 $\overline{K_3}$ free

Consider a graph with independence number $\alpha(G) \leq 2$. A very natural characterization is that $\overline{G}$ is Δ-free. So, G is $\overline{K_3}$ free. So for a graph with MVS G_{xy} such that $\alpha(G_{xy}) \leq 2$, implies G_{xy} doesn't contain $\overline{K_3}$. Now we shall find out all minimal forbidden graphs for such graphs.

The simplest class of minimal forbidden graph with this property is shown in Figure 4.1, which is a theta. Now we add chords. Other edges that aren't chords cannot be added since it violates the minimality condition. We can do this in two ways: keeping the chord in one side of $\overline{K_3}$; or using one of the vertices of $\overline{K_3}$ to form the chord. We cannot add an edge across the $\overline{K_3}$, else it won't be the MVS anymore. So we first add the edge on one side of $\overline{K_3}$.

Suppose we add one chord forming a cycle C_n such that $x \in C_n$ and $n > 3$, as shown in Figure 4.2. Here we can do some minimality transformations, depending

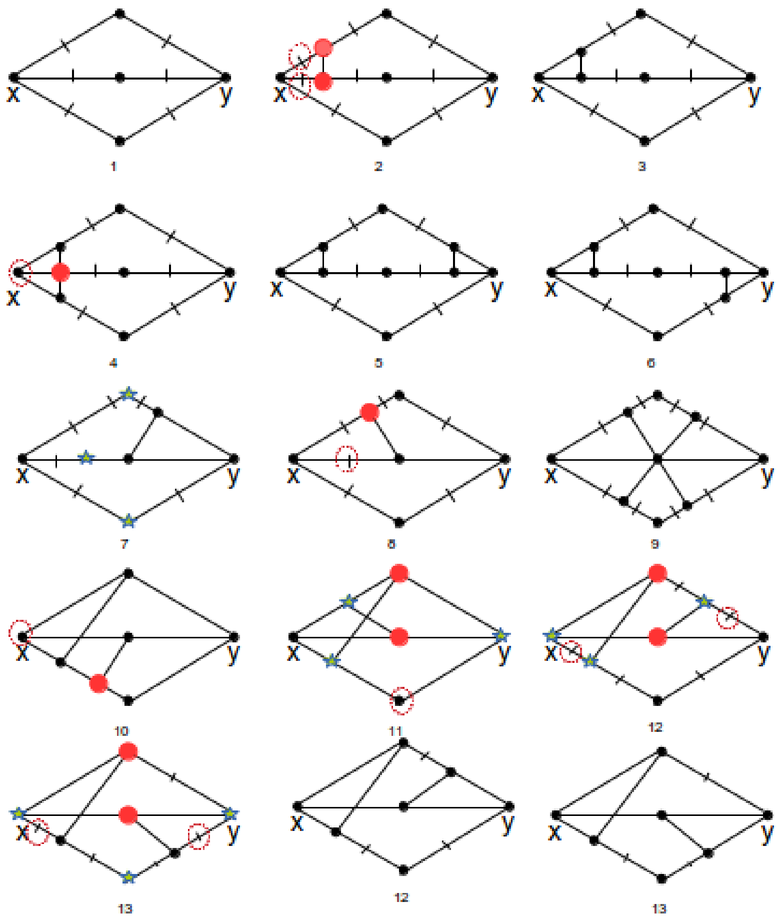


Figure 4: $\overline{K_3}$ Analysis

upon which of the highlighted vertex we shift x to. Hence the only minimal forbidden graph in this case will have a triangle C_3, forming a pyramid as shown in Figure 4.3. No C_n are allowed for $n > 3$. So we can just add edges forming triangles. This can be done in thre ways: first form the Δ in the same side of $\overline{K_3}$ as the first Δ or form Δ's on the opposite side of $\overline{K_3}$ (which can be done in two ways). In the first case the minimality conditions aren't satisfied, as shown in Figure 4.4. The second and third cases are shown in Figure 4.5 and 4.6 which are isomorphic to each other, forming a prism.

Now let's consider the second case where the edge is added to one of the vertices of $\overline{K_3}$. Now *wlog* let us add an edge to the cental vertex of $\overline{K_3}$ to the outer cycle. If we allow the central xy path to have a length greater than 2, we get Figure 4.7, which is isomorphic to a pyramid. Now we restrict the central xy path to have a length 2, else the minimal conditions aren't satisfied (ref Figure 4.8). In such a case we can add any number of chords, maintaining the $\overline{K_3}$. This will result in a wheel like structure containing an induced $\overline{K_3}$ i.e. 0-wheel, as shown in Figure 4.9.

Now let's consider new edges involving two vertices of $\overline{K_3}$. There are two possibilities worth considering(it can be easily checked that others violate the minimality conditions): both edges end in the same side of $\overline{K_3}$, or they end in different sides. In the first case again there are two possibilities, but the minimality conditions are violated in both of them (ref. Figure 4.10 and 4.11). The second case also has two possibilities as shown in Figure 4.12 and 4.13, ehich after a minimality transformation turn out to be a prism as shown in Figure 4.14 and 4.15 respectively. (We analyse how this configuration is reached in appendix.) If we consider new edges involving three vertices of $\overline{K_3}$, it is easy to see that Figure 4.10 and 4.11 will be a subgraph, so it violates the minimality conditions.

Now we try to form hybrid structures by combining cases discussed above. The only possible structure that can be formed is to include a triangle in a wheel. However it is easy to see that the structures so formed are isomorphic to pyramid. This concludes our analysis. So the set of minimal forbidden graphs are shown in Figure 5.

A.1.4 $K_1 \bigcup K_2$ free

The class of complete k-partite graph $K_{n_1,n_2,..}$ is $K_1 \bigcup K_2$ free. So graphs with MVS as complete k-partite graphs don't have $K_1 \bigcup K_2$.

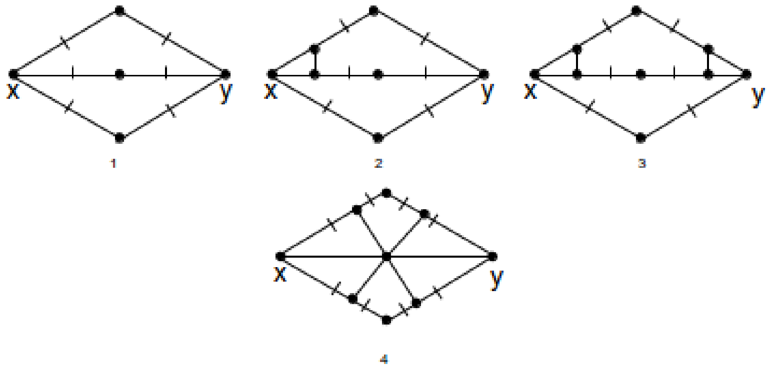


Figure 5: $\overline{K_3}$-free Forbidden graphs

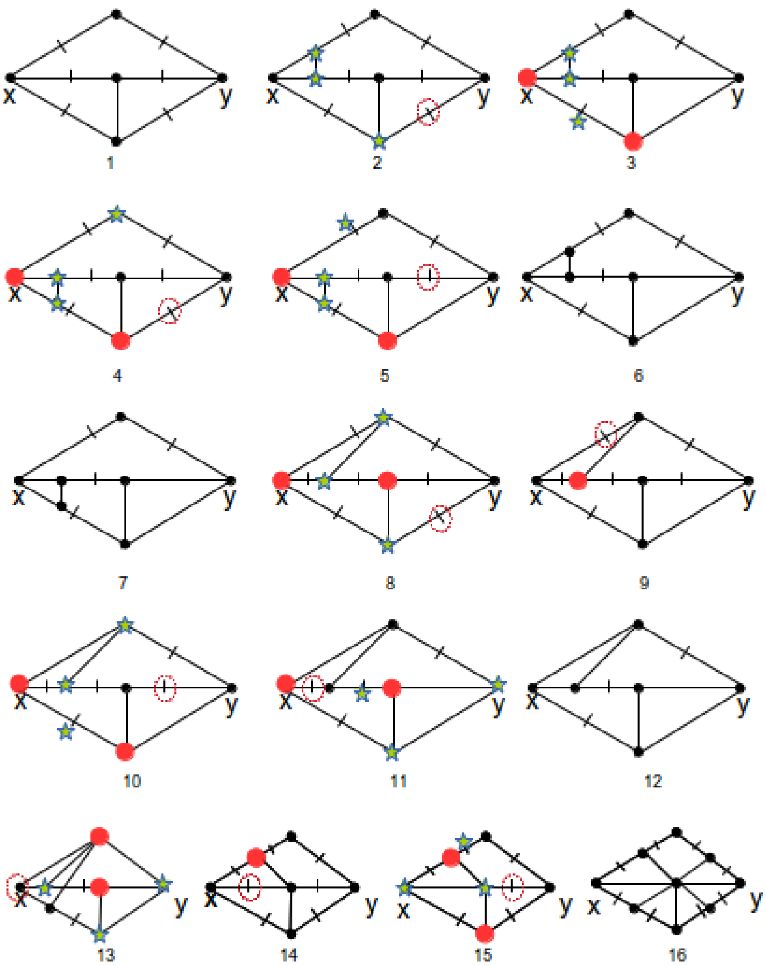


Figure 6: $K_1 \bigcup K_2$ Analysis

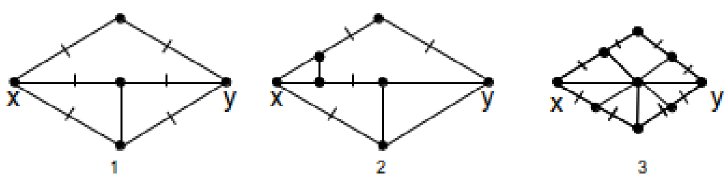


Figure 7: $K_1 \bigcup K_2$-free Forbidden graphs

The simplest class of minimal forbidden graphs that has $K_1 \bigcup K_2$-free MVS i.e. 1-theta is shown in Figure 6.1. Now we add chords. Other paths that aren't chords cannot be added since it violates the minimality conditions. This can be done in two ways: involving the vertices other that that of $K_1 \bigcup K_2$; and involving the vertices of $K_1 \bigcup K_2$. In the first case we don't allow chords to cross across $K_1 \bigcup K_2$ else it ceases to be MVS.

Analysing similar to the last section, we can see that the only minimal graph in this case will have a triangle C_3. Yet on appropriately selecting MVS $K_1 \bigcup K_2$ as shown in Figure 6.2 to 6.5, we can do a minimality transformation to get Figure 6.6 and 6.7 respectively which results in a prism. If we try to add another chord, again as in previous section, we can't add a Δ in the same side, also here we can't add a Δ on the opposite side of $K_1 \bigcup K_2$ (due to unavailability of vertices on the central xy path).

Now let us consider the case where one of the vertices of $K_1 \bigcup K_2$ is involved in the chord. This again can happen in two ways; first involves the K_1, and the other involves K_2. So we draw a chord from K_1 to one of xy paths. It goes through a series of minimality transformations as shown in Figure 6.8 to 6.11 to give Figure 6.12. Additional edges cannot be added since it violates the minimality conditions as shown in Figure 6.13. Now we involve vertices of K_2, *wlog* it is sufficient to analyse one vertex. After going through a few minimality transformations (Figure 6.14 and 6.15), we reach the wheel inducing $K_1 \bigcup K_2$ (Figure 6.16) i.e. 1-wheel. Now we try to form some hybrid structures, but it can be easily checked that no such structures can be formed. This concludes our analysis and the set of minimal forbidden graphs are shown in Figure 7.

A.1.5 P_3 free

The class of complete k-partite graph $K_{n_1,n_2,..}$ is P_3 free. So graphs with MVS as complete k-partite graphs don't have P_3.

The simplest class of minimal forbidden graphs that has P_3-free MVS i.e. 2-theta is shown in Figure 8.1. Now we add chords. Other paths that aren't chords cannot be added since it violates the minimality conditions. This can be done in two ways: involving the vertices other that that of P_3; and involving the vertices of P_3. In the first case we don't allow chords to cross across P_3 else it ceases to be MVS.

Analysing similar to the last two sections, we can see that the only minimal graph in this case will have a triangle C_3. So we get a 2-pyramid as shown in

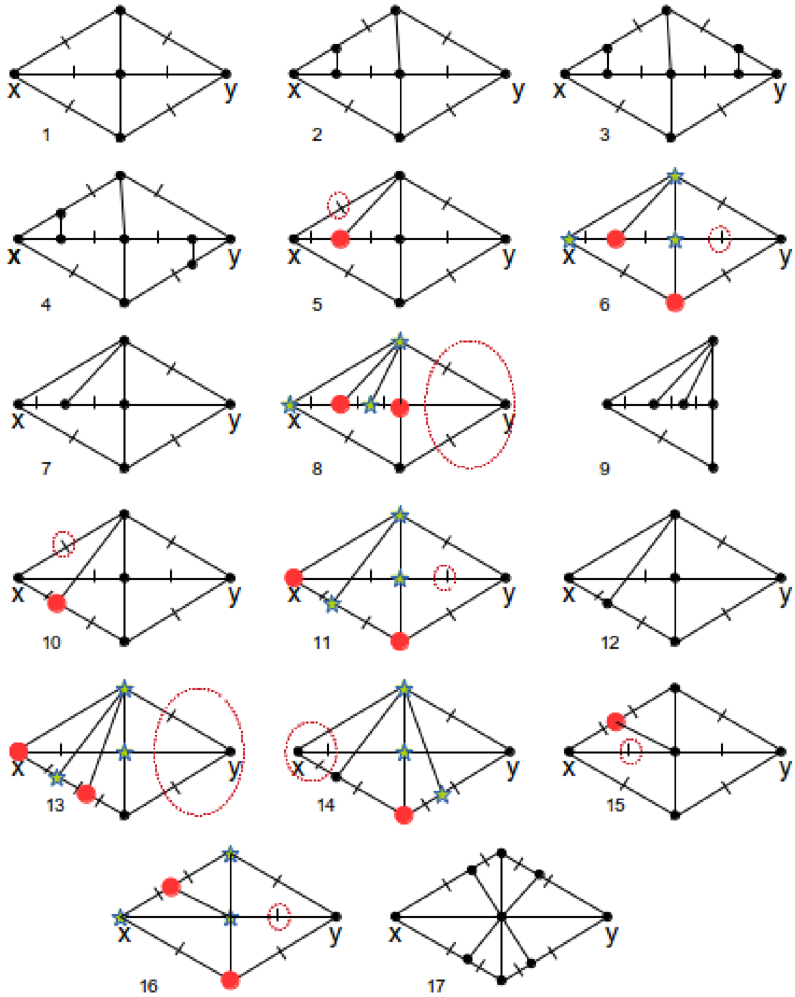


Figure 8: P_3 Analysis

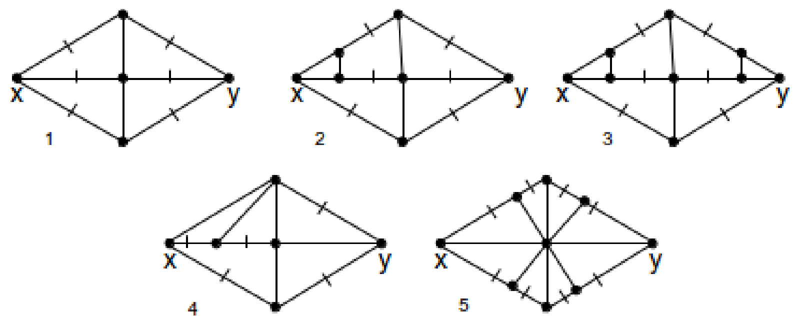


Figure 9: P_3-free Forbidden graphs

Figure 8.2. If we try to add another chord, again as in previous sections, we can't add a Δ in the same side, so we add in opposite side resulting in two isomorphic structures of 2-prisms as shown in Figures 8.3 and 8.4.

Now let us consider the case where one of the vertices of P_3 is involved in the chord. This again can happen in two ways; first involves the end vertices, and the other involves the interior vertex. So we draw a chord from an end vertex to the middle xy path going through the interior vertex of P_3. It goes through a series of minimality transformations as shown in Figure 8.5 and 8.6 to give Figure 8.7 which is a theta with P_4. Now if we add more such edges we can do another minimality transformation as shown in Figure 8.8 to get a wheel (Figure 8.9). Now we draw a chord from the end vertex to the outer xy path going through the other end vertex of P_3. It goes through a series of minimality transformations as shown in Figure 8.10 and 8.11 to give Figure 8.12, which is isomorphic to Figure 8.7 hence is a theta with a P_4. Now if we add more such edges we can do another minimality transformation as shown in Figure 8.13 to 8.16 to get a 2-wheel(Figure 8.17) which is obtained when we draw edges from the interior vertex of P_3 after going through structures shown in Figure 8.15 and 8.16.

Now we try to form some hybrid structures, but it can be easily checked that no such structures can be formed. This concludes our analysis and the set of minimal forbidden graphs are shown in Figure 9.

A.1.6 K_3 free

The simplest class of minimal forbidden graphs that has K_3-free MVS i.e. 3-theta is shown in Figure 10.1. Now we add chords. Other paths that aren't chords cannot be added since it violates the minimality conditions. This can be done in two ways: involving the vertices other that that of K_3; and involving the vertices of K_3. In the first case we don't allow chords to cross across K_3 else it ceases to be MVS.

Analysing similar to the last three sections, we can see that the only minimal graph in this case will have a triangle C_3. So we get a 3-pyramid as shown in Figure 10.2. If we try to add another chord, again as in previous sections, we can't add a Δ in the same side, so we add in opposite side resulting in two isomorphic structures i.e. 3-prisms as shown in Figures 10.3 and 10.4.

Now let us consider the case where one of the vertices of K_3 is involved in the chord. Due to symmetry, we can consider one vertex. It goes through a series

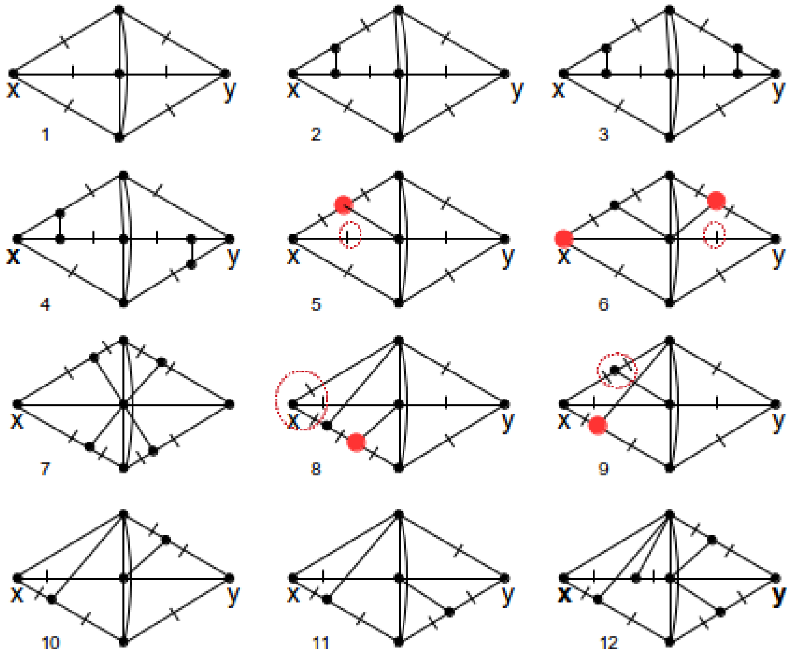


Figure 10: K_3 Analysis

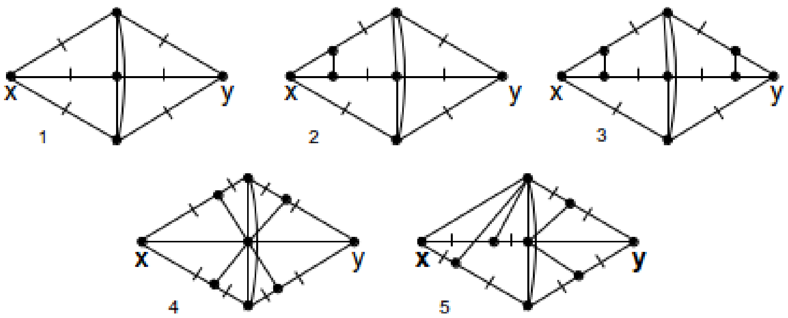


Figure 11: K_3-free Forbidden graphs

of minimality transformations as shown in Figure 10.5 and 10.6 and on adding edges gives a 1-co-wheel as shown in Figure 10.7. Now we consider edges from two vertices of K_3. This can happen in four ways as shown in Figure 10.8 to 10.11, out of which two don't satisfy the minimality conditions. On adding more edges we get a 2-co-wheel, as shown in Figure 10.12.

Now we try to form some hybrid structures, but it can be easily checked that no such structures can be formed. This concludes our analysis and the set of minimal forbidden graphs are shown in Figure 11. $\square$