Insertions for feasibility of clustered trees on grid intersection graphs

Thesis

Noam Malter

Supervisors: Dr. Nili Beck and Prof. Michal Stern

May 29, 2022

Contents

1 Introduction

Analyzing organizational networks is a fairly new area, that started from social networks analysis and has been gaining momentum in recent years. There are many insights that can be learned about the organization from analyzing of the work relations network and its hierarchy. The findings can provide insights about the organizational culture, cooperation between employees and departments, internal cooperation within departments and more. The idea is to consider the organizational network as a graph, where people working in the various departments of the organization are represented as vertices, and groups of people assigned to certain projects are represented as clusters. Our goal is to enable a fast and secure flow of information inside each cluster. Therefore, we demand that each subgraph induced by a cluster has to be a subtree. When the given network does not have a feasible solution tree, adding a vertex to a cluster is interpreted as adding an employee to the particular project. Allowing such additions will gain a structure of a tree, such that each cluster will be represented by a subtree in the solution tree.

The problem of finding whether each subgraph induced by a cluster is a subtree is characterized by the Clustered Spanning Tree problem. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, where $V = \{v_1, \ldots, v_n\}$ a set of vertices and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of not necessarily disjoint clusters $S_i \subseteq V, 1 \leq i \leq m$. The Clustered Spanning Tree problem, denoted CST, aims to find, given, in a complete graph induced on the vertices of V, whether there exists a spanning tree, such that each cluster induces a subtree.

Another possible application comes from the area of bioinformatics. An evolutionary tree is a tree graph showing the evolutionary relationships among various biological species or other entities (their phylogeny), based upon similarities and differences in their physical or genetic characteristics. Each vertex in the tree represents one of the species, and each cluster represents a common feature, e.g. a shared gene or protein. The problem is to find the evolutionary tree, in this case a directed tree, under the constraint that each cluster creates a connected subtree. Note that in these trees, the leaves represent species that exist today, while some of the internal vertices cannot be directly observed. When no evolutionary tree exists, using the solution we propose, inserting vertices to clusters, means adding a certain attribute to one of the species, which is very likely for unknown species. The purpose of the insertions is to find a consistent evolutionary tree with consistency of all species described by all clusters.

An important and essential question is whether a feasible solution tree exists for a given instance of the CST problem. There are two conditions that are required for a hypergraph to have a feasible solution tree. The hypergraph should satisfy the Helly property, and its intersection graph should be chordal. A hypergraph satisfies the Helly property if, for any subset of clusters, if the intersection between every two clusters is not empty then all clusters in the subset contain at least one common vertex. A chordal graph is a graph in which all cycles of four or more vertices have a chord, which is an edge that is not part of the cycle but connects two vertices of the cycle.

If no feasible solution tree exists, we consider inserting vertices to clusters from $\mathcal{S}$ in order to gain feasibility, by finding feasible vertices insertion lists with minimum cardinality. We find a minimum cardinality feasible vertices insertion list by looking at the intersection graph of H, for specific types of hypergraphs.

Example 1.1. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph with $V = \{v_1, \ldots, v_4\}$ a set of vertices and clusters $\{S_1, S_2, S_3, S_4\}$, where $S_1 = \{v_1, v_2\}$, $S_2 = \{v_2, v_4\}$, $S_3 = \{v_1, v_3\}$, $S_4 = \{v_3, v_4\}$. Hypergraph H is described in Figure 1.

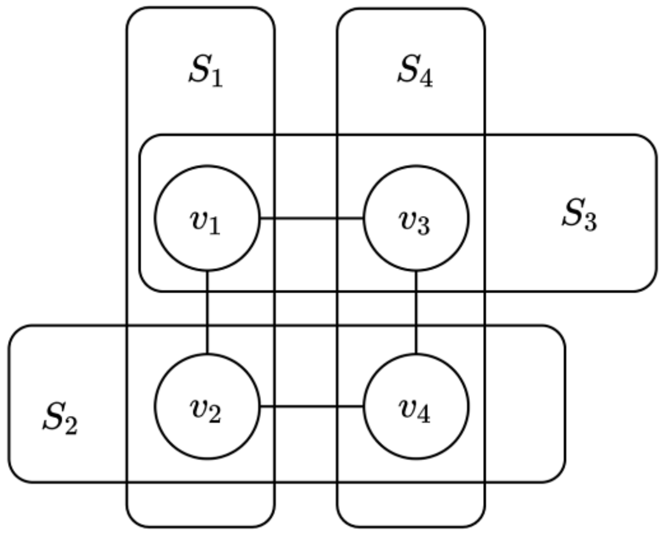


Figure 1: Hypergraph H

It is obvious that for every cluster which contains exactly two vertices, an edge connecting the vertices is required in any solution tree. Therefore, the solution tree for H is the one presented in Figure 2. Clearly, this is not a tree as it contains cycle $v_1 - v_3 - v_4 - v_2 - v_1$. Hence, H has no feasible solution tree.

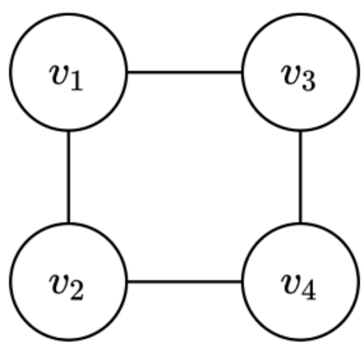


Figure 2: No solution tree for H

However, consider the following vertices insertion list $IL = \{(v_2, S_3), (v_4, S_3)\}$. Figure 3, describes the new hypergraph created after inserting the vertices in IL to the corresponding clusters in H.

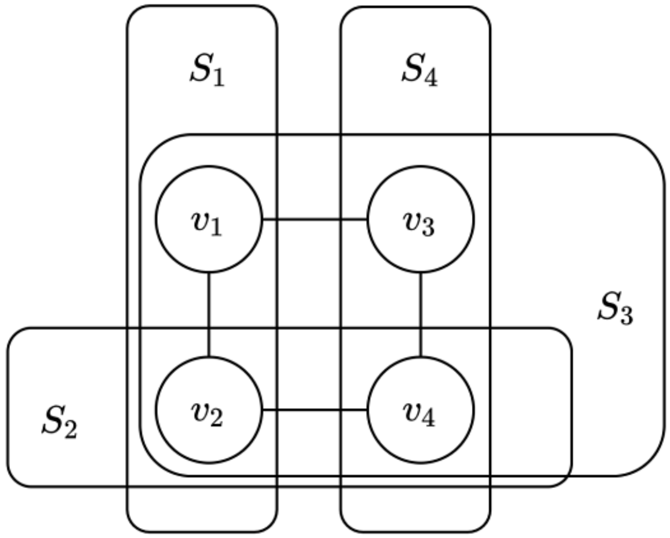


Figure 3: Hypergraph H after inserting the vertices from IL

Figure 4 describes a possible solution tree for H. We can see that the solution tree spans all vertices in V, and each cluster induces a subtree.

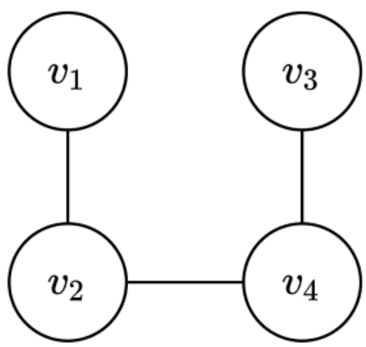


Figure 4: A possible solution tree for H

There are not many studies dealing with CST problem, where no feasible solution tree exists. In a previous work, Guttmann-Beck, Sorek and Stern [6] characterized when inserting vertices to exactly one cluster attains feasibility, for those instances where no feasible solution tree exists. This approach finds the appropriate cluster and the vertices that should be inserted.

Levin [9], focused on cases where the intersection graph of H is a d-level t-domino graph, see definition 4.12. An algorithm is provided that finds a possible feasible vertices insertion list for the intersection graph of H, with minimum cardinality.

Our research considers specific types of hypergraphs, namely $n \times m$ grid hypergraphs, which are hypergraphs where the intersection graph is constructed of chordless cycles with size four, where every two cycles intersect in at most one edge, see Figure 5. We denote the nodes of the intersection graph $\{s_{0,0}, \ldots, s_{n,0}, \ldots s_{0,m}, \ldots, s_{n,m}\}$, corresponding to the clusters set $\mathcal{S} = \{S_{0,0}, \ldots, S_{n,0}, \ldots S_{0,m}, \ldots, S_{n,m}\}$. Since the intersection graph is chordless there is no feasible solution tree. Therefore, we consider on adding chords between nodes of the intersection graph in order to gain feasibility, by finding feasible chords addition lists with minimum cardinality.

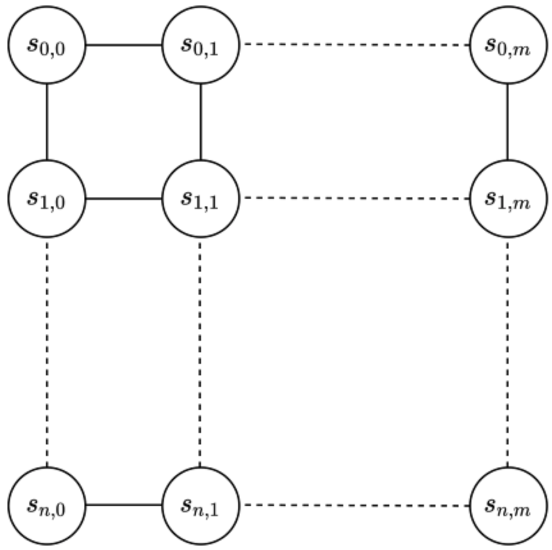


Figure 5: An $n \times m$ grid graph

One main result of this research is a method called Convert to Clique, which is a feasible vertices insertion list that can be easily implemented for any graph. We show that in some cases a feasible vertices insertion list constructed using the Convert to Clique method is also a minimum cardinality feasible vertices insertion list. Other important results are minimum cardinality feasible vertices insertion lists for $n \times 1$ and $n \times 2$ grid hypergraphs.

In a related work, Guttmann-Beck, Rozen and Stern [5], presented another version of achieving feasibility by removing vertices from clusters in order to gain feasibility. They provide an algorithm which finds a possible list of vertices removal, which creates a new hypergraph with a feasible solution tree, and an algorithm which finds the corresponding solution tree, thus proving the correctness of the list of vertices removal.

An important known and most restricted case of CST problem is where the solution tree is required to be a path, such that every cluster induces a subpath in the solution path. A solution to this problem is testing for the Consecutive Ones Property, denoted by COP. A binary matrix has the COP when there is a permutation of its rows that gains the ones to be consecutive in every column. Booth and Lueker [1] introduced a data structure called a PQ-tree. PQ-trees can be used to represent the permutations of V in which the vertices of each cluster are required to occur consecutively.

Florescu [4] focused on finding feasible solution trees by removing or inserting a minimum number of vertices from or into the clusters of the given hypergraphs. The research focuses on cases where the intersection graph has a specific shape, specifically, triangular base shapes, such as a diamond and a butterfly, and also consider cactus tree intersection graphs and triangle free intersection graphs.

The Feasibility Clustered Travelling Salesman Problem, denoted by $FCTSP$, is to verify whether there exists a simple path that visits each vertex exactly once, such that the vertices of each cluster are visited consecutively. Sayag [11] focused on hypergraphs with not necessarily disjoint clusters, where there is no feasible solution of $FCTSP$. For those instances with no feasible solution, the research investigates the removal of vertices from clusters, in order to achieve a feasible solution for the new set of clusters. The research presents several algorithms which find a removal list of vertices from appropriate clusters, in order to gain feasibility. The research investigates special and different characteristics of the given hypergraph and its intersection graph, and considers structures of graph families for the intersection graph, including intersection graphs that are simple paths, chordless cycles, trees, stars, caterpillar trees, bipartite graphs, cliques and graphs that contain a cut edge or a cut node.

While researching the CST problem, we were seeking a way to achieve chordality of intersection graphs. Thus, we encountered the Chords for Chordality problem, denoted by CFC, which is to find chords (edges) addition lists whose addition achieves chordality in G. This problem is a possible chordal completion. A chordal completion of a given graph G is a chordal graph, on the same vertex set V, that has G as a subgraph. A minimal chordal completion is a chordal completion such that any graph formed by removing an edge would no longer be a chordal completion. A different type of chordal completion, one that minimizes the size of the maximum clique in the resulting chordal graph, can be used to define the treewidth of G.

This work is organized as follows. Section 2 introduces definitions that will be used throughout the paper. Section 3 introduces CST and CFC problems, showing that a minimum cardinality feasible vertices insertion list is not necessarily a minimum cardinality feasible chords addition list and vice versa. Section 4 introduces general lemmas and Convert to Clique method. It also generalizes results related to d-level t-domino graphs. Section 5 discusses vertices insertion lists providing minimum cardinality vertices insertion lists for specific hypergraphs and introducing internal and external Helly vertices and outside insertions. Section 6 discusses chords addition lists, providing minimum cardinality chords addition lists for 2×1, 3×1, 4×1 one sided clique grid graphs, a generic method for constructing a feasible chords addition list for an $n \times 1$ one sided clique grid graph, where n is even. Furthermore, expressing the minimum cardinality of a chords addition list using linear programming. Section 7 discusses summary and further research.

2 Definitions

In this paper, we research the Clustered Spanning Tree problem, denoted by CST and the Chords for Chordality problem, denoted by CFC. Therefore, we provide the following definitions for the problems as well as examples.

Definition 2.1. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, where $V = \{v_1, \ldots, v_n\}$ a set of vertices and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of not necessarily disjoint clusters $S_i \subseteq V, 1 \leq i \leq m$. The Clustered Spanning Tree problem, denoted by CST, aims to find whether, in a complete graph induced on the vertices of V, there exists a spanning tree, such that each cluster induces a subtree.

Example 2.2. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph with $V = \{v_1, \ldots, v_4\}$ a set of vertices and clusters $\{S_1, S_2, S_3\}$, where $S_1 = \{v_1, v_2\}$, $S_2 = \{v_2, v_3\}$, $S_3 = \{v_3, v_4\}$. Hypergraph H is described in Figure 6.

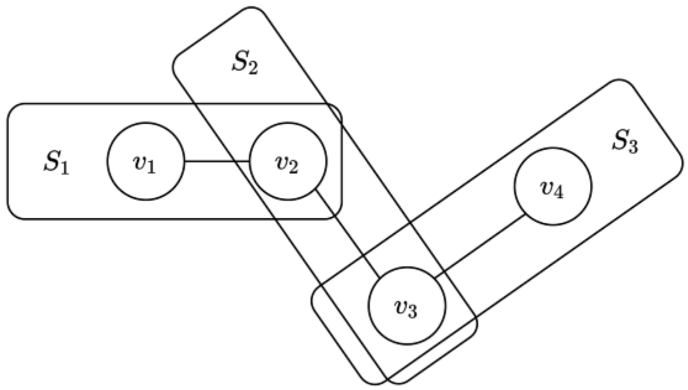


Figure 6: Hypergraph H

A possible solution tree for H is the tree described in Figure 7. We can see that the solution tree spans all vertices in V, and each cluster induces a subtree.

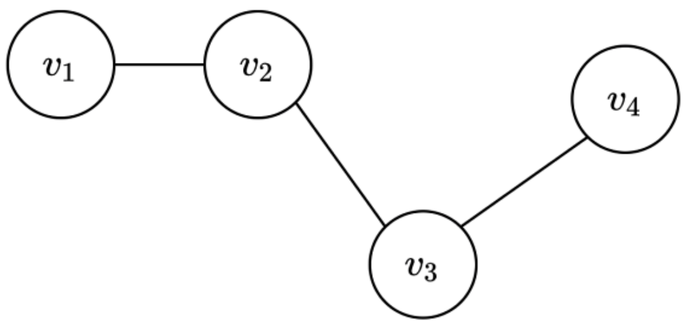


Figure 7: A possible solution tree for H

Definition 2.3. Let $G = \langle U, E \rangle$ be a graph. The Chords for Chordality problem, denoted by CFC, aims to find chords (edges) addition lists whose addition achieves chordality in G.

Example 2.4. Let $G = \langle U, E \rangle$ be a graph with $U = \{s_1, \ldots, s_4\}$ a set of vertices and $E = \{(s_1, s_2), (s_1, s_3), (s_2, s_4), (s_3, s_4)\}$ a set of edges. The graph G is described in Figure 8.

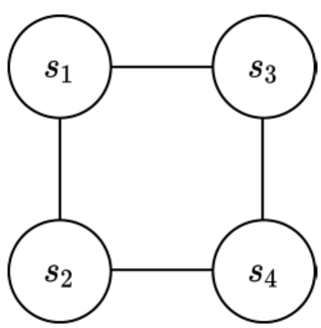


Figure 8: Graph G

Clearly, G is chordless as the cycle $s_1 - s_3 - s_4 - s_2 - s_1$ exists. A possible chords addition list is

$AL = \{(s_2, s_3)\}$. Figure 9 describes $G + AL$, a new instance of the graph after adding the chord from AL, colored red. The new instance of the graph is chordal.

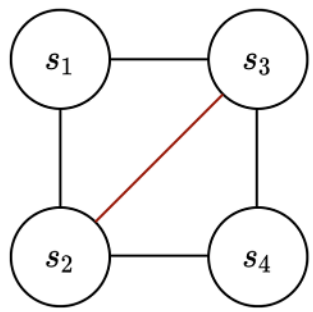


Figure 9: Graph $G + AL$

A restricted problem of CST is the Clustered Spanning Tree by Paths. The following definition introduces the problem.

Definition 2.5. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, where $V = \{v_1, \ldots, v_n\}$ a set of vertices and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of not necessarily disjoint clusters $S_i \subseteq V, 1 \leq i \leq m$. The Clustered Spanning Tree by Paths problem, denoted by $CSTP$, aims to decide whether there exists a path-based tree support, which is a tree spanning the vertices of V, such that each cluster induces a path.

Example 2.6. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph with $V = \{v_1, \ldots, v_7\}$ a set of vertices and clusters $\{S_1, S_2, S_3, S_4\}$, where $S_1 = \{v_1, v_2, v_3\}$, $S_2 = \{v_1, v_4, v_5\}$, $S_3 = \{v_1, v_6, v_7\}$, $S_4 = \{v_1, v_2, v_4\}$.

Hypergraph H is described in Figure 10.

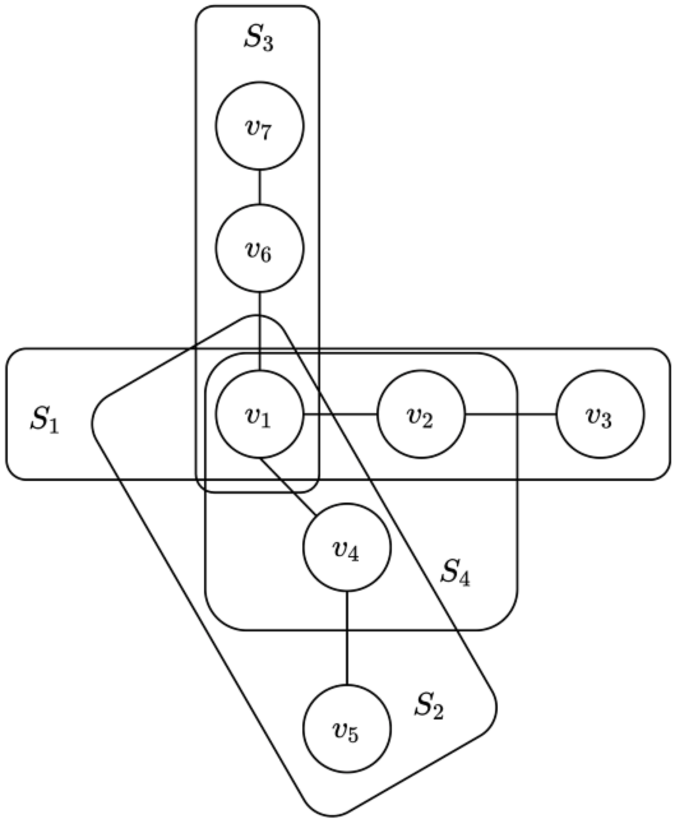


Figure 10: Hypergraph H

A possible solution tree for H is the tree described in Figure 11. We can see that the solution tree spans all vertices in V, and each cluster induces a path.

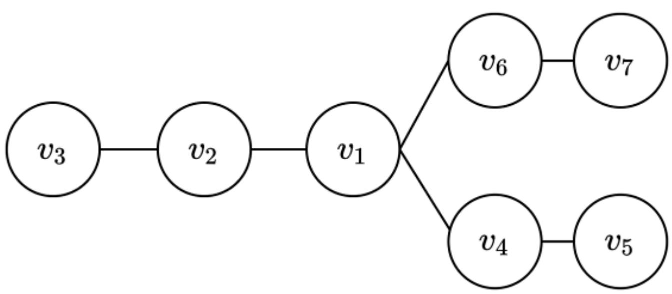


Figure 11: A possible solution tree for H

In this work, we consider the intersection graph of a hypergraph H in order to determine if it is chordal. Therefore, we provide the following definition for an intersection graph.

Definition 2.7. Given a hypergraph $H = \langle V, \mathcal{S} \rangle$, where V is a set of vertices and $\mathcal{S}$ is a set of not necessarily disjoint subsets $\{S_1, \ldots, S_p\}$ of V called clusters. The **intersection graph** of $\{S_1, \ldots, S_p\}$, denoted by $\boldsymbol{G_{int}(\{S_1, \ldots, S_p\})}$, is defined to be a graph whose set of nodes is $\{s_1, \ldots, s_p\}$ where s_i corresponds to S_i, and an edge (s_i, s_j) exists whenever $S_i \cap S_j \neq \emptyset$.

In order to gain feasibility for the CST problem, we add vertices to clusters in a hypergraph H. Therefore, we provide the following definition for a vertices insertion list.

Definition 2.8. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph. $IL = \{(v_1, S_{i_1}), \ldots, (v_k, S_{i_k})\}$ is a **Vertices Insertion list** of H if IL is a list of pairs where $v_j \notin S_{i_j}$, such that inserting every vertex v_j to cluster S_{i_j} creates a new instance of the hypergraph denoted by $H + IL$. The intersection graph of $H + IL$ is $G_{int}(\mathcal{S} + IL)$. If the new hypergraph $H + IL$ has a feasible solution tree we say that IL is a **feasible vertices insertion list of** H.

There are many vertices insertion lists that gain feasibility for the CST problem for a given hypergraph H. However, we seek a vertices insertion list whose cardinality is minimal. The following definition highlights our main goal of achieving minimum cardinality of an insertion list.

Definition 2.9. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph. We define $\mathbf{mIL(H)} = min\{\,|IL| \,:\, IL$ is a feasible vertices insertion list $\}$. IL' is a **Minimum Cardinality Feasible Vertices Insertion List** for H if IL' is a feasible vertices insertion list of H and $|IL'| = mIL(H)$.

In order to achieve chordality in a graph U, we add chords to the graph. The following definitions introduce chords addition lists and the minimum cardinality version of this list.

Definition 2.10. Let $G = \langle U, E \rangle$ be a graph. $AL = \{(s_{i_1}, s_{j_1}), \ldots, (s_{i_k}, s_{j_k})\}$ is a **Chords Addition list** of G if AL is a list of chords (edges), such that adding every chord (s_{i_1}, s_{j_1}) creates a new instance of the graph denoted by $G + AL$. If the new graph $G + AL$ is chordal we say that AL is a **feasible chords addition list of** H.

Definition 2.11. Let $G = \langle U, E \rangle$ be a graph. We define $\mathbf{mAL(G)} = min\{\,|AL| \,:\, AL$ is a feasible chords addition list $\}$. AL' is a **Minimum Cardinality Feasible Chords Addition List** for G if AL' is a feasible chords addition list of G and $|AL'| = mAL(G)$.

In our research we focus sometimes on subproblems of a given instance and the corresponding induced graph.

Definition 2.12. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph. Let $\mathcal{S}' \subseteq \mathcal{S}$ be a set of clusters. We define $\mathbf{H}[\mathcal{S}']$ to be the hypergraph whose **Vertex Set** is $\mathbf{V}(\mathcal{S}') = \bigcup_{S_i \in \mathcal{S}'} S_i$ and its clusters set is $\mathcal{S}$'. The **induced graph** $G_{int}(\mathcal{S})[\{S_i \,|\, S_i \in \mathcal{S}\}]$ is the intersection graph of $H[\mathcal{S}']$ and therefore can be denoted by $\mathbf{G_{int}}(\mathcal{S}')$. If $IL = \{(v_0, S_0), \ldots, (v_j, S_k)\}$ is a vertices insertion list and $\mathcal{S}' \subseteq \mathcal{S}$, we define the **Induced Vertices Insertion List** to be $IL[\mathcal{S}'] = \{(v, S_i) \mid (v, S_i) \in IL, S_i \in \mathcal{S}'\}$.

Definition 2.13. Let $G = \langle U, E \rangle$ be a graph. Let $U' \subseteq U$ be a set of nodes. We define $\mathbf{G[U']}$ the induced graph on U' to be the graph whose nodes set is U' and edge set is $\{(u_1, u_2) \,|\, u_1 \in U', u_2 \in U', (u_1, u_2) \in E\}$.

3 Clustered Spanning Tree and Chords For Chordality

An important and essential question in our research is whether a given hypergraph has a feasible solution tree. As stated in Theorem 3.2, a necessary considition is that the intersection graph is required to be chordal for achieving feasibility for the CST problem. Since we were interested in achieving a minimum cardinality feasible insertion list, we studied whether this implies a minimum cardainality chords addition list for the corresponding intersection graph. However, the simple example of a 2×2 grid hypergraph presented in this section demonstrates that the minimum solutions for the two problems do not necessarily coexist.

We start by introducing criterias for the existance of a feasible solution tree for a given hypergraph. First, we will need the following definition.

Definition 3.1. Let $\mathcal{S} = \{S_1, \ldots, S_p\}$ be a family of subsets. We say that $\mathcal{S}$ satisfies the **Helly Property** if the following holds: For every $\mathcal{S}' \subseteq \mathcal{S}$, if each pair members of $\mathcal{S}'$ intersect, then all the members of $\mathcal{S}'$ have a common element. In other words, if every $S_i, S_j \in \mathcal{S}'$ satisfy $S_i \cap S_j \neq \emptyset$ then $\bigcap_{S_i \in \mathcal{S}'} S_i \neq \emptyset$.

The following theorem summarizes the conditions for feasibility.

Theorem 3.2. (Duchet [2], Flament [3], Slater [12]) A hypergraph $H = \langle V, \mathcal{S} \rangle$ has a feasible solution tree if and only if it satisfies the Helly property and its intersection graph is chordal.

Example 3.3. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph with $V = \{1, \ldots, 12\}$ a set of vertices and clusters $\{S_{0,0}, S_{0,1}, S_{0,2}, S_{1,0}, S_{1,1}, S_{1,2}, S_{2,0}, S_{2,1}, S_{2,2}\}$, where $S_{0,0} = \{1, 2\}$, $S_{0,1} = \{1, 3, 4\}$, $S_{0,2} = \{3, 5\}$, $S_{1,0} = \{2, 6, 7\}$, $S_{1,1} = \{4, 6, 8, 9\}$, $S_{1,2} = \{5, 8, 10\}$, $S_{2,0} = \{7, 11\}$, $S_{2,1} = \{9, 11, 12\}$, $S_{2,2} = \{10, 12\}$. The intersection graph $G_{int}(\mathcal{S})$ is described in Figure 12.

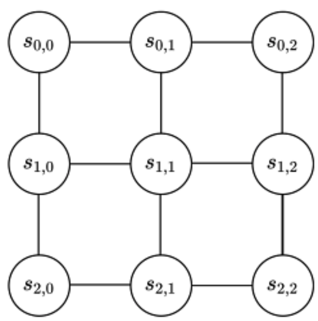


Figure 12: The intersection graph $G_{int}(\mathcal{S})$ of example 3.3

Consider the following chords addition list: $AL = \{(s_{1,0}, s_{0,1}), (s_{1,0}, s_{2,1}), (s_{0,1}, s_{1,2}), (s_{2,1}, s_{1,2}), (s_{0,1}, s_{2,1})\}$.

In Lemma 6.5, we will prove that this list is a minimum cardinality feasible chords addition list. Figure 13 shows $G_{int}(\mathcal{S}) + AL$.

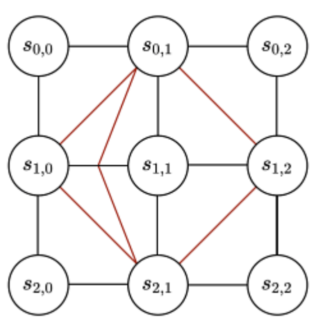


Figure 13: A 2×2 grid graph with the chords of AL

While AL solves the CFC problem, according to McKee and McMorris [10], another necessary condition for CST is that the hypergraph has to satisfy Helly property as stated in Theorem 3.2. Consider the following vertices insertion list: $IL = \{(1, S_{1,0}), (4, S_{1,0}), (9, S_{1,0}), (11, S_{1,0}), (3, S_{1,2}), (9, S_{1,2}), (12, S_{1,2}), (4, S_{2,1})\}$. IL is a feasible vertices insertion list of H such that $G_{int}(\mathcal{S} + IL) = G_{int}(\mathcal{S}) + AL$ and such that $H + IL$ satisfies Helly property.

After inserting IL to H, the clusters are: $S_{0,0} = \{1, 2\}$, $S_{0,1} = \{1, 3, 4\}$, $S_{0,2} = \{3, 5\}$, $S_{1,0} = \{2, 6, 7, 1, 4, 9, 11\}$, $S_{1,1} = \{4, 6, 8, 9\}$, $S_{1,2} = \{5, 8, 10, 3, 9, 12\}$, $S_{2,0} = \{7, 11\}$, $S_{2,1} = \{9, 11, 12, 4\}$, $S_{2,2} = \{10, 12\}$.

Note that the red vertices are new vertices inserted to clusters by IL. Furthermore, $|IL| = 8$. However, consider the following vertices insertion list: $IL' = \{(1, S_{0,2}),\ (1, S_{1,0}),\ (1, S_{1,1}),\ (1, S_{1,2}),\ (1, S_{2,0}),\ (1, S_{2,1}),\ (1, S_{2,2})\}$. It is clear that $|IL'| = 7$. In Theorem 5.26, we will prove that IL' is a minimum cardinality vertices insertion list.

We note that $G_{int}(\mathcal{S}+IL')$ is a clique on nine nodes. Therefore, the number of edges in $G_{int}(\mathcal{S}+IL')$ is $\binom{9}{2} = 36$. The number of edges in $G_{int}(\mathcal{S})$ is 12. Therefore, the cardinality of the corresponding chords addition list is 24.

Thus, a minimum cardinality feasible chords addition list of $G_{int}(\mathcal{S})$ does not correspond to a minimum cardinality feasible vertices insertion list of H. Following the above discussion we will present results for both problems.

4 Clustered Spanning Tree

In this section, we consider various hypergraphs and study feasible vertices insertion lists for those hypergraphs. We introduce a method called "Convert to Clique" to construct a feasible vertices insertion list and prove it is a minimum vertices insertion list of certain hypergraphs. We present general lemmas regarding CST.

4.1 General Lemmas

This section contains general lemmas which will be used throughout the paper. The following lemma is an important result regarding Clustered Spanning Tree by Paths, denoted $CSTP$. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, where $V = \{v_1, \ldots, v_n\}$ a set of vertices and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of not necessarily disjoint clusters $S_i \subseteq V, 1 \leq i \leq m$. The Clustered Spanning Tree by Paths problem, denoted by $CSTP$, is to find whether a tree spanning all vertices in V exists, such that each cluster induces a path. Since $CSTP$ is a restricted case of CST, the result in the following lemma is true for CST as well.

The lemma and its proof are presented by Guttmann-Beck and Stern [7].

Lemma 4.1. Consider a hypergraph $H = \langle V, \mathcal{S} \rangle$. If T is a feasible solution tree for $CSTP$ problem and X is an intersection of a set of clusters from $\mathcal{S}$, then $T[X]$ is a connected path.

More important results that will be used later are presented and proved in the following lemmas.

Lemma 4.2. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, with $\mathcal{S} = \{S_1, \ldots, S_m\}$. Let $\mathcal{S}' = \{S_{i_1}, \ldots, S_{i_k}\} \subseteq \mathcal{S}$, and let IL be a feasible vertices insertion list of H, then $IL[\mathcal{S}']$ is a feasible vertices insertion list of $H[\mathcal{S}']$.

Proof. Since IL is a feasible vertices insertion list of H, hypergraph $H + IL$ has a feasible solution tree, denote this tree T. The clusters in $H + IL$ are $\{S_1 \cup IL(\{S_1\}), \ldots, S_m \cup IL(\{S_m\})\}$. The clusters in $H[\mathcal{S}'] + IL[\mathcal{S}']$ are $\{S_{i_1} \cup IL(\{S_{i_1}\}), \ldots, S_{i_k} \cup IL(\{S_{i_k}\})\}$. Since T is a feasible solution tree, $T[\mathcal{S}']$ is chordless and spans a tree for every cluster in $S_{i_1} \cup IL[S_{i_1}], \ldots, S_{i_k} \cup IL[S_{i_k}]$, and can be connected by arbitrary edges to create a feasible solution tree for $\mathcal{S}'$. Therefore, $IL[\mathcal{S}']$ is a feasible vertices insertion list of $H[\mathcal{S}']$. □

Definition 4.3. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph. If H satisfies the Helly property, $v \in V$ is called a **Helly vertex** if there is $\mathcal{S}' \subseteq \mathcal{S}$ such that $G_{int}(\mathcal{S})$ induces a clique on $\mathcal{S}'$ and $v \in S$ for every $S \in \mathcal{S}'$.

Lemma 4.4. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph and let IL be a feasible vertices insertion list of H. If IL contains a pair (v^*, S^*) such that v^* is not a Helly vertex in $H + IL$, then $IL \setminus (v^*, S^*)$ is also a feasible vertices insertion list of H.

Proof. Since IL is a feasible vertices insertion list, according to Theorem 3.2, $G_{int}(\mathcal{S} + IL)$ is chordal and satisfies the Helly property. If $v^* \in S^*$ then clearly $IL \setminus (v^*, S^*)$ is a feasible vertices insertion list. Suppose $v^* \notin S^*$. Denote $IL' = IL \setminus (v^*, S^*)$. Let $\mathcal{S}' \subseteq \mathcal{S}$ be the list of clusters in $H + IL$ which contains v^* excluding S^*, $\mathcal{S}' = \{S | S \in \mathcal{S} \setminus \{S^*\}, v^* \in S \cup IL[S]\}$. For every $S \in \mathcal{S}'$ the intersection graph $G_{int}(\mathcal{S} + IL)$ contains a chord from S to S^*. Therefore, $G_{int}(\mathcal{S} + IL)$ contains a clique on $\mathcal{S}' \cup \{S^*\}$, denote this clique as K. Since $H + IL$ satisfies Helly property, there is $v \in V, v \neq v^*$, such that v is a Helly vertex which corresponds to K. Therefore, $v \in [S^* \cup IL[S^*]] \cap [S \cup IL[S]]$ for every $S \in \mathcal{S}'$. Thus, $G_{int}(\mathcal{S} + IL')$ still contains K. Therefore, removing (v^*, S^*) from the vertices insertion list, does not remove any chord from $G_{int}(\mathcal{S} + IL)$, thus $G_{int}(\mathcal{S} + IL) = G_{int}(\mathcal{S} + IL')$, and therefore, $G_{int}(\mathcal{S} + IL')$ is chordal. Since v^* is not a Helly vertex, $H + IL'$ satisfies Helly property. Hence, IL' is a feasible vertices insertion list. □

Notation 4.5. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, IL a vertices insertion list of H which includes a pair (v, S') and another vertex $u \in V$. We denote $\boldsymbol{IL_{v \to u}}$ to be the new vertices insertion list created by replacing every appearance of v in IL by u. $IL_{v \to u} = IL \setminus \{(v, S) \mid S \in \mathcal{S}_v\} \cup \{(u, S) \mid S \in \mathcal{S}_v\}$ where $\mathcal{S}_v = \{S \mid S \in \mathcal{S}, (v, S) \in IL\}$.

Remark 4.6. $|IL_{v \to u}| \leq |IL|$ since for every pair (u, S) which was added to the list, we also removed a pair (v, S). It may be strictly smaller if there is a cluster S such that $(u, S) \in IL$. Hence, if IL is a minimum cardinality feasible vertices insertion list, so is $IL_{v \to u}$.

Definition 4.7. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph and $\mathcal{S}' \subseteq \mathcal{S}$. A vertices insertion list IL is called an **Inner Vertices Insertion List** with respect to $\mathcal{S}'$ if every pair $(v, S) \in IL$ satisfies $v \in V(\mathcal{S}')$ and $S \in \mathcal{S}'$.

Lemma 4.8. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, let IL be a feasible vertices insertion list of H and let $\mathcal{S}' \subseteq \mathcal{S}$. There exists a feasible inner vertices insertion list with respect to $\mathcal{S}'$ denoted by IL', such that $|IL'| \leq |IL[\mathcal{S}']|$.

Proof. Since IL is a feasible vertices insertion list of H, according to Lemma 4.2, $IL[\mathcal{S}']$ is a feasible vertices insertion list of $H[\mathcal{S}']$, and let T' be the corresponding feasible solution tree. Consider a pair $(v, S) \in IL[\mathcal{S}']$ such that $S \in \mathcal{S}'$ and $v \in V, v \notin V(\mathcal{S}')$. Let $\{S_{i_1}, \ldots, S_{i_k}\}$ be the list of clusters in $\mathcal{S}'$ such that v was inserted to them according to $IL[\mathcal{S}']$. Let $U = (S_{i_1} \cup IL[S_{i_1}]) \cap (S_{i_2} \cup IL[S_{i_2}]) \cap \ldots \cap (S_{i_k} \cup IL[S_{i_k}])$. According to Lemma 4.1, $T'[U]$ is a connected subtree which contains vertex v. Let u be a neighbor of v in T' such that u is a vertex in at least one of the clusters $(S_{i_1} \cup IL[S_{i_1}]), (S_{i_2} \cup IL[S_{i_2}]), \ldots, (S_{i_k} \cup IL[S_{i_k}])$. Without loss of generality, suppose that $u \in S_{i_1} \cup IL[S_{i_1}]$ and possibly other clusters. Let $IL' = IL[\mathcal{S}']_{v \to u}$. IL' is created from $IL[\mathcal{S}']$ by removing pairs $(v, S_{i_1}), (v, S_{i_2}), \ldots, (v, S_{i_k})$ and adding pairs $(u, S_{i_2}), (u, S_{i_3}), \ldots, (u, S_{i_k})$. Therefore, $|IL'| \leq |IL[\mathcal{S}']| - 1$. We claim that IL' is a feasible vertices insertion list of $H[\mathcal{S}']$. Remove from T' all the edges which connect v to neighbors in $\mathcal{S}'$, excluding u. Note that edge (v, u) remains in T'. Denote by T'' the new subgraph. T'' is connected, and the number of edges in T'' equals to the number of edges in T' and therefore T'' is a tree. Every cluster $S \in \mathcal{S} \setminus \mathcal{S}'$ remains connected, as $T'[S] = T''[S]$. For $S \in \mathcal{S}'$, $T''[S]$ remains connected by using u instead of v. Continue to change $IL[\mathcal{S}']$ in this manner until it does not contain any pair $(v, \mathcal{S}), v \notin V(\mathcal{S}')$. According to Remark 4.6 $|IL_{v \to u}| \leq |IL|$. In this case, $IL_{v \to u} = IL'$ and therefore, $|IL'| \leq |IL[\mathcal{S}']|$. □

4.2 Convert to a Clique

In this section, we describe a method which achieves chordality in a hypergraph, for all possible hypergraphs, by inserting a vertex from the intersection of two clusters to all other clusters of the hypergraph. Thus, converting the intersection graph into a clique.

Convert to Clique method is:

- Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph.
- Choose $v \in S_i \cap S_j$ for $S_i \neq S_j, S_i \in \mathcal{S}, S_j \in \mathcal{S}$.
- Define $IL(v) = \{(v, S) | S \neq S_i, S \neq S_j, S \in \mathcal{S}\}$.

Theorem 4.9. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph. Let $IL(v)$, for $v \in S_i \cap S_j$, be a vertices insertion list constructed by Convert to Clique method. $IL(v)$ is a feasible vertices insertion list and $|IL(v)| = |\mathcal{S}| - 2$.

Proof. To prove IL is a feasible vertices insertion list, we need to show that $G_{int}(\mathcal{S} + IL)$ is chordal and that Helly property is satisfied. It is clear that $G_{int}(\mathcal{S} + IL)$ is a clique as we add v to every cluster in H. Hence, $G_{int}(\mathcal{S} + IL)$ is chordal. In addition, since $v \in S$ for every $S \in \mathcal{S}$, Helly property is satisfied. Furthermore, by Convert to Clique method it is trivial that $|IL(v)| = |\mathcal{S}| - 2$. □

Remark 4.10. A feasible solution tree of a hypergraph H, with a vertices insertion list IL constructed by the Convert to Clique method, is a star shaped tree with v as its center.

Example 4.11. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph with clusters $\{S_{0,0}, S_{0,1}, S_{0,2}, S_{1,0}, S_{1,2}, S_{2,0}, S_{2,1}, S_{2,2}\}$, where $S_{0,0} = \{1, 2\}$, $S_{0,1} = \{1, 3\}$, $S_{0,2} = \{3, 4\}$, $S_{1,0} = \{2, 5\}$, $S_{1,2} = \{4, 6\}$, $S_{2,0} = \{5, 7\}$, $S_{2,1} = \{7, 8\}$, $S_{2,2} = \{6, 8\}$. The intersection graph $G_{int}(\mathcal{S})$ is described in Figure 14.

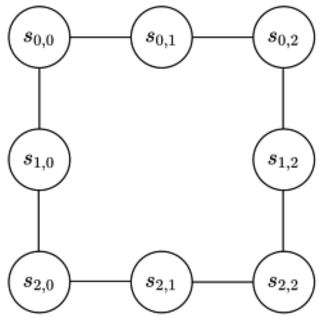


Figure 14: The intersection graph $G_{int}(\mathcal{S})$ of example 4.11

We construct a vertices insertion list using the Convert to Clique method. We arbitrarily choose $v = S_{0,0} \cap S_{0,1} = 1$, $IL = \{(1, S_{0,2}), (1, S_{1,0}), (1, S_{1,2}), (1, S_{2,0}), (1, S_{2,1}), (1, S_{2,2})\}$. After inserting IL to H, the intersection graph $G_{int}(\mathcal{S} + IL)$ is described in Figure 15.

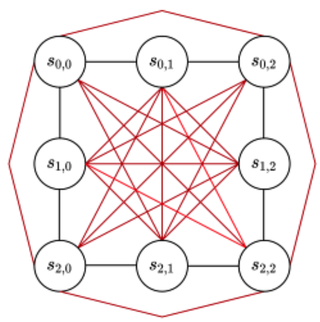


Figure 15: The intersection graph $G_{int}(\mathcal{S} + IL)$

The solution tree of $H + IL$, is described in Figure 16.

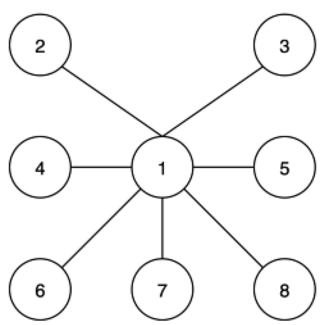


Figure 16: $H + IL$ solution tree

4.3 Feasible Addition Edges List

The following section contains results from the work presented by Levin [9]. His work presented a minimum cardinality feasible vertices insertion lists for a special family of hypergraphs, a d-level t-domino graph which will be presented in the following definition. In this section, we prove that a vertices insertion list created by the Convert to Clique method is also a minimum cardinality feasible vertices insertion list for these hypergraphs.

Definition 4.12. A **d-level t-domino**, is a family of graphs that has d uncontained chordless cycles, denoted by $C_1, \ldots, C_d$, for $d \geq 1$. The graph satisfies:

- $C_i \cap C_{i+1}$ for $i = 1, \ldots, d-1$, is a path which contains t nodes, for $t \geq 2$.
- $C_i \cap C_j = \emptyset$ if $|i-j| > 1$ for $i, j = 1, \ldots, d$.
- $|C_i| \geq 2t$, for $i = 1, \ldots, d$.

Define the nodes in $C_i \cap C_{i+1}$ **path nodes** and denote them by $s_{i,k}$, for $i = 1, \ldots, d-1$ and $k = 1, \ldots, t$, representing clusters $S_{i,k}$. All the other nodes are denoted as regular nodes $r_{i,k}$, for $i = 1, \ldots, d$ and $k = 1, \ldots, K_i$, where cycle C_i contains K_i regular nodes, representing clusters $R_{i,k}$, see Figure 17.

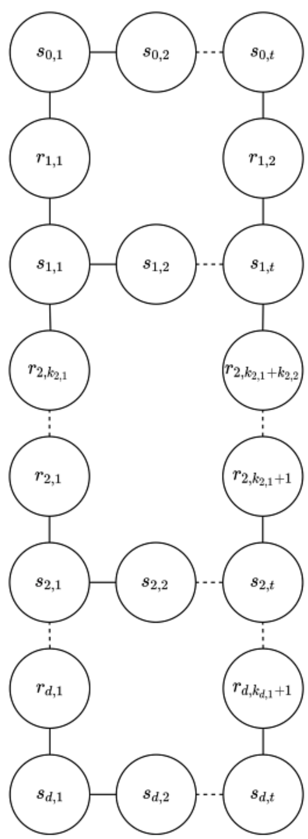


Figure 17: A d-level t-domino graph

Definition 4.13. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph whose intersection graph $G_{int}(\mathcal{S})$ is a d-level t-domino graph and whose cycles are $C_1, C_2, \ldots, C_d$. Define **FAEL**, Feasible Addition Edges List, to be a chords addition list which contains the following edges:

- $(s_{i,1}, s_{i,j})$, for $i = 0, \ldots, d$, $j \geq 3$.
- $(s_{i,1}, r_{i,p_i})$, for $i = 1, \ldots, d$ and $p_i = 2, \ldots, k_i$.

- $(s_{i,1}, s_{i-1,1})$, for $i = 1, \ldots, d$.
- $(s_{i,1}, s_{i-1,j})$, for $i = 1, \ldots, d, j \geq 3$.

Note that $G_{int}(\mathcal{S})$ contains edge $(s_{i,1}, r_{i,1})$, for every $i = 1, \ldots, d$.

Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph whose intersection graph $G_{int}(\mathcal{S})$ is a d-level t-domino graph. Levin [9] presents a feasible vertices insertion list with cardinality $|L_{FAEL}| = \sum_{i=1}^{d}(|C_i|) - dt + t - 2$. This calculation assumes that at least one regular node exists in every cycle. Levin [9] proves that this list is a minimum cardinality feasible vertices insertion list of H. We generalize the definition and calculation of $FAEL$ for the case the d-level t-domino graph may have different sizes of intersection between cycles.

4.3.1 FAEL for constant size of intersections

Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph whose intersection graph is a d-level t-domino graph. The number of nodes in $G_{int}(\mathcal{S})$ is: $|\mathcal{S}| = \sum_{i=1}^{d}(|C_i|) - (d-1)t$, as there are $(d-1)t$ nodes which belong to exactly two cycles. Since Convert to Clique method requires $|\mathcal{S}| - 2$ vertices, we get: $|L_{FAEL}| = |\mathcal{S}| - 2 = \sum_{i=1}^{d}(|C_i|) - (d-1)t - 2 = \sum_{i=1}^{d}(|C_i|) - dt + t - 2$. Therefore, L_{FAEL} and a vertices insertion list achieved by using the Convert to Clique method, have the same cardinality and both are minimum cardinality vertices insertion lists.

4.3.2 Generalized FAEL calculation

The following is a simple generalization of results presented by Levin [9], for the case where the intersection between cycle C_i and cycle C_{i+1} is of size t_i and f_i indicates whether there exists at least one regular node in C_i.

Remark 4.14. When the size of the intersection of two adjacent cycles is not constant, we denote $t_i = |C_i \cap C_{i+1}|, t_0 = 0, t_d = 0$.

$$f_i = \begin{cases} 1 & \text{a regular node } r \text{ exists on the left side of a cycle at level } i \\ 0 & \text{otherwise} \end{cases}$$

The first equation is based on the work presented by Levin [9].

$$\begin{aligned}
|L_{FAEL}| &= \sum_{i=0}^{d}(t_i - 2) + \sum_{i=1}^{d}\left[|C_i| - (t_i + t_{i-1}) + 1 - f_i\right] + \sum_{i=1}^{d} f_i + d \\
&= \sum_{i=0}^{d}(t_i - 2) + \sum_{i=1}^{d}|C_i| - \sum_{i=1}^{d} t_i - \sum_{i=1}^{d}(t_{i-1}) + d - \sum_{i=1}^{d} f_i + \sum_{i=1}^{d} f_i + d \\
&= \sum_{i=0}^{d} t_i - 2(d+1) + \sum_{i=1}^{d}|C_i| - \sum_{i=1}^{d} t_i - \sum_{i=0}^{d-1} t_i + 2d \\
&= \sum_{i=0}^{d} t_i - 2(d+1) + \sum_{i=1}^{d}|C_i| - \sum_{i=0}^{d} t_i - \sum_{i=1}^{d-1} t_i + 2d \\
&= \sum_{i=1}^{d}|C_i| - \sum_{i=1}^{d-1} t_i - 2(d+1) + 2d \\
&= \sum_{i=1}^{d}|C_i| - \sum_{i=1}^{d-1} t_i - 2d - 2 + 2d \\
&= \sum_{i=1}^{d}|C_i| - \sum_{i=1}^{d-1} t_i - 2
\end{aligned}$$

Consider the last part of the equation: $\sum_{i=1}^{d-1} t_i$ is the number of nodes which belong to exactly two cycles in $G_{int}(\mathcal{S})$. Therefore, $\sum_{i=1}^{d}|C_i| - \sum_{i=1}^{d-1} t_i$ is the number of nodes in $G_{int}(\mathcal{S})$ which is also $|\mathcal{S}|$. Since Convert to Clique method requires $|\mathcal{S}| - 2$ vertices insertions, we get that the cardinality of the list is $\sum_{i=1}^{d}|C_i| - \sum_{i=1}^{d-1} t_i - 2$. Thus, the number of vertices insertions in L_{FAEL} and in Convert to Clique method is identical.

5 Hypergraphs with $n \times m$ grid intersection graph

In this section we discuss hypergraphs whose intersection graph is an $n \times m$ grid. We start by introducing basic cycles and grids. Most of our work focuses on researching inserting vertices into clusters for hypergraphs whose intersection graph is an $n \times m$ grid and adding chords to graphs that are an $n \times m$ grid graph.

5.1 $n \times m$ grids

In this section we introduce definitions and a general lemma relevant for $n \times m$ hypergraphs.

Definition 5.1. Let $G = \langle V, E\rangle$ be a graph. A cycle in G is a non-empty path in which the only repeated vertices are the first and last vertices. A **Basic Cycle** is a four edges path in which the first vertex is equal to the last vertex. A basic cycle is chordless and contains four vertices.

Definition 5.2. An $n \times m$ **grid graph** is a family of graphs that has n rows and m columns of uncontained four node chordless cycles, defined basic cycles, and denoted by $C_{1,1}, C_{1,2}, \ldots, C_{1,m}, C_{2,1}, \ldots, C_{n,m}$. A basic cycle at level i, j, for $1 \leq i < n, 1 \leq j < m$, contains nodes $\{s_{i-1,j-1}, s_{i,j-1}, s_{i-1,j}, s_{i,j}\}$. The graph satisfies:

- $C_{i,j} \cap C_{i+1,j}$, for every $i \in \{1, \ldots, n-1\}$ and $j \in \{1, \ldots, m\}$, is a path which contains two nodes.
- $C_{i,j} \cap C_{i,j+1}$, for every $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m-1\}$, is a path which contains two nodes.
- $C_{i,j} \cap C_{k,l} = \emptyset$, if $|i-k| > 1$ or $|j-l| > 1$, for every $i, k \in \{1, \ldots, n\}$ and $j, l \in \{1, \ldots, m\}$.

See Figure 18.

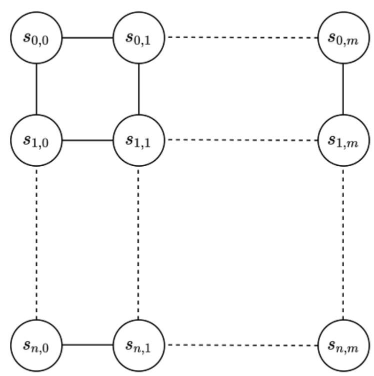


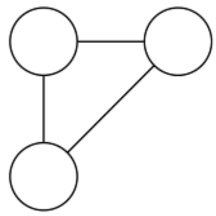
Figure 18: An $n \times m$ grid graph

Note that the numbering of nodes in a grid graph starts from 0 to n for the rows and from 0 to m for the columns.

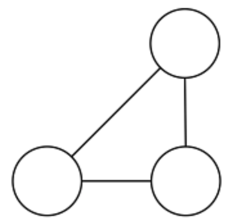
Property 5.3. The number of nodes in an $n \times m$ grid graph is $(n+1)(m+1)$.

Definition 5.4. A hypergraph whose intersection graph is an $n \times m$ grid graph is $n \times m$ **grid hypergraph**. The clusters in an $n \times m$ grid hypergraph are $\mathcal{S} = \{S_{0,0}, S_{1,0}, \ldots, S_{n,0}, S_{0,1}, S_{1,1}, \ldots, S_{n,m}\}$.

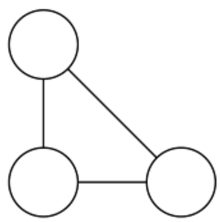
Definition 5.5. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times m$ grid hypergraph, and let IL be a feasible vertices insertion list. Δ is a **Basic Triangle** if Δ is a triangle in $G_{int}(\mathcal{S} + IL)$ such that the three nodes of Δ reside in one basic cycle of $G_{int}(\mathcal{S})$. There are four possible shapes for a basic triangle, as described in Figure 19.



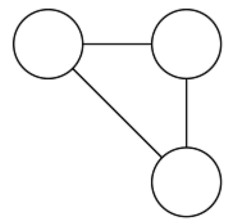
(a) A basic triangle



(b) A basic triangle



(c) A basic triangle



(d) A basic triangle

Figure 19: Basic triangles

Lemma 5.6. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph whose intersection graph is an $n \times m$ grid. Let IL be a feasible vertices insertion list of H. For every cluster $S_i \in \mathcal{S}$, at least one of the followings hold:

1. $\exists v \in V \setminus S_i$ such that $(v, S_i) \in IL$. This means that at least one vertex is inserted to S_i by IL.
2. $\exists v \in S_i$ and $S \neq S_i$ such that $(v, S) \in IL$. This means that a vertex from S_i is inserted to another cluster by IL.

Proof. Since IL is a feasible vertices insertion list, according to Theorem 3.2, $G_{int}(\mathcal{S}+IL)$ is chordal. Since $G_{int}(\mathcal{S})$ is an $n \times m$ grid, every node in $G_{int}(\mathcal{S} + IL)$ is part of a clique which contains at least three nodes. Let K be a clique which contains s_i and let s', s'' be two nodes, different from s_i, which are nodes in K. Let S', S'' be the clusters which correspond to nodes s' and s''. Since IL is a feasible vertices insertion list, according to Theorem 3.2, $H + IL$ satisfies the Helly property. Since K is a clique in $G_{int}(\mathcal{S}+IL)$, there is a vertex $v \in [S_i \cup IL(S_i)] \cap [S' \cup IL(S')] \cap [S'' \cup IL(S'')]$. Since $G_{int}(\mathcal{S})$ is an $n \times m$ grid which does not contain any triangles, any vertex $v \in V$ belongs to at most two clusters in $\mathcal{S}$. Therefore, $v \notin S_i$ or $v \notin S'$ or $v \notin S''$. If $v \in S_i$ then $(v, S_i) \in IL$ and statement 1 is satisfied, otherwise $v \notin S_i$. If $v \notin S'$ then $(v, S') \in IL$ and statement 2 is satisfied. If $v \notin S''$ then $(v, S'') \in IL$ and statement 2 is satisfied. $\square$

5.2 Vertices Insertions

In this section we find minimum cardinality feasible vertices insertion lists for $n \times 2$ grid hypergraphs.

5.2.1 Minimum vertices insertion lists for $n \times 1$ grid hypergraphs

Consider $n \times 1$ grid hypergraphs whose intersection graph is described in Figure 20.

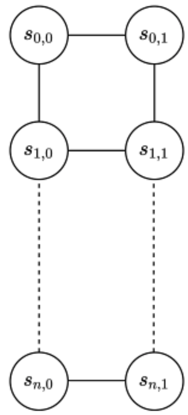


Figure 20: The intersection graph of an $n \times 1$ grid hypergraph

We start with the most basic and trivial case of a grid hypergraph whose intersection graph is a 1×1 grid.

Lemma 5.7. Let $H = \langle V, \mathcal{S} \rangle$ be a 1×1 grid hypergraph. Let $\mathcal{S} = \{S_{0,0}, S_{1,0}, S_{0,1}, S_{1,1}\}$. If IL is a feasible vertices insertion list of H, then $|IL| \geq 2$.

Proof. $G_{int}(\mathcal{S})$ contains one basic cycle C. According to Theorem 3.2, if IL is a feasible vertices insertion list, $G_{int}(\mathcal{S}+IL)$ is chordal. Therefore, C requires at least one chord to achieve chordality. There are two options to add a chord to C, $(s_{0,0}, s_{1,1})$ or $(s_{0,1}, s_{1,0})$. Each option divides C into two basic triangles Δ_1 and Δ_2. Furthermore, according to Theorem 3.2, $H + IL$ satisfies Helly property. Each triangle represents a set of clusters that requires at least one Helly vertex. By the structure of H there is no vertex that belongs to more than two clusters. Therefore, at least two insertions are required. □

Lemma 5.8. Let $H = \langle V, \mathcal{S} \rangle$ be a 1×1 grid hypergraph. Let $\mathcal{S} = \{S_{0,0}, S_{1,0}, S_{0,1}, S_{1,1}\}$. There exists IL which is a feasible vertices insertion list of H such that $|IL| = 2$.

Proof. The following are four possible options for a feasible vertices insertion list:

- $IL_1 = \{(v, s_{1,1}), (u, s_{1,1})\}, v \in S_{0,0} \cap S_{0,1}, u \in S_{0,0} \cap S_{1,0}$.
- $IL_2 = \{(v, s_{1,0}), (u, s_{1,0})\}, v \in S_{0,0} \cap S_{0,1}, u \in S_{0,1} \cap S_{1,1}$.
- $IL_3 = \{(v, s_{0,0}), (u, s_{0,0})\}, v \in S_{0,1} \cap S_{1,1}, u \in S_{1,0} \cap S_{1,1}$.
- $IL_4 = \{(v, s_{0,1}), (u, s_{0,1})\}, v \in S_{0,0} \cap S_{1,0}, u \in S_{1,0} \cap S_{1,1}$.

We will prove that IL_1 is a feasible vertices insertion list. $G_{int}(\mathcal{S} + IL_1)$ is a chordal hypergraph combined of two triangles $\Delta_1 = \{s_{0,0}, s_{0,1}, s_{1,1}\}$ and $\Delta_2 = \{s_{0,0}, s_{1,0}, s_{1,1}\}$. The clusters corresponding to the nodes in triangles Δ_1 and Δ_2 share a common vertex v and u respectively. Hence, $H + IL$, satisfies Helly property. Thus, IL_1 is a feasible vertices insertion list of H and $|IL| = 2$. A similar proof holds also for IL_2, IL_3, IL_4. □

Theorem 5.9. Let $H = \langle V, \mathcal{S} \rangle$ be a 1×1 grid hypergraph. $mIL(H) = 2$.

Proof. According to Lemma 5.7, $mIL(H) \geq 2$. The cardinality of the vertices insertion lists constructed in Lemma 5.8 is 2. Therefore, $mIL(H) = 2$. $\square$

Now we consider a more general case of a grid hypergraph whose intersection graph is an $n \times 1$ grid. The following definition will be used in the lemma that follows it.

Definition 5.10. (v_1, v_t) is **separating edge** of a connected graph $G = (V, E)$ if G contains an edge (v_1, v_t) and by removing both vertices v_1 and v_t from G disconnects G into two connected components, whose vertex sets are V_a, V_b such that $V = V_a \cup V_b \cup \{v_1, v_t\}$ with $V_a \cap V_b = \emptyset$. However, G remains connected if we remove only v_1 or v_t.

Theorem 5.11. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph. $mIL(H) = 2n$.

Proof. This proof is by induction on n, the number of cycles in the graph. For $n = 1$, according to Theorem 5.9, $mIL(H) = 2$. Assume the lemma is correct for an $(n-1) \times 1$ grid hypergraph, and prove for $n \times 1$ grid hypergraph. Edge $(s_{n-1,0}, s_{n-1,1})$ is a separating edge in $G_{int}(\mathcal{S})$, see Figure 21, and it splits $G_{int}(\mathcal{S})$ into a $(n-1) \times 1$ grid hypergraph and 1×1 grid hypergraph. Note that both sides contain the separating edge. Denote by $\mathcal{S}_a$ the clusters corresponding to the nodes in the $(n-1) \times 1$ grid hypergraph and $\mathcal{S}_b$ the clusters corresponding to the nodes in the 1×1 grid hypergraph. According to Guttmann-Beck and Stern [7], $mIL(H) = mIL(H[\mathcal{S}_a]) + mIL(H[\mathcal{S}_b])$. From the induction assumption, $mIL(H[\mathcal{S}_a]) = 2(n-1)$. According to Theorem 5.9, $mIL(H[\mathcal{S}_b]) = 2$. Hence, $mIL(H) = mIL(H[\mathcal{S}_a]) + mIL(\mathcal{S}_b) = 2(n-1) + 2 = 2n$. $\square$

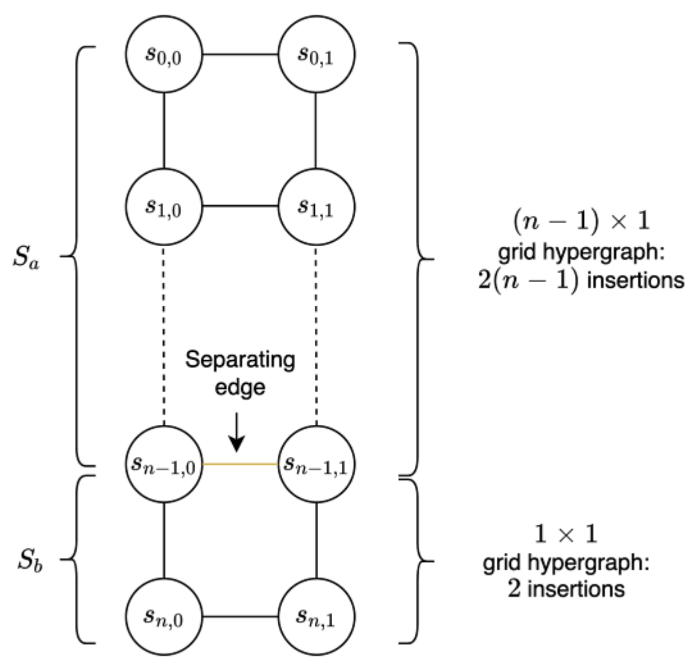


Figure 21: A visual description of proof of Theorem 5.11

Theorem 5.12. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph. Let IL be a vertices insertion list constructed by Convert to Clique method. IL is a minimum cardinality feasible vertices insertion list.

Proof. According to Theorem 4.9, IL is a feasible vertices insertion list with cardinality $|\mathcal{S}| - 2$. According to property 5.3, $|\mathcal{S}| = (n+1)(1+1) = 2n+2$, and therefore, $|IL| = 2n$. According to Theorem 5.11, IL is a minimum cardinality feasible vertices insertion list. $\square$

5.2.2 Internal and External Helly vertices

In this section we define inner and external Helly vertices and discuss how these vertices affect the cardinality of a feasible vertices insertion list.

Definition 5.13. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, IL be a feasible vertices insertion list of H, v is a Helly vertex and $v \in V(\mathcal{S})$, then v is an **Inner Helly vertex**.

Definition 5.14. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, IL be a feasible vertices insertion list of H, v is a Helly vertex and $v \notin V(\mathcal{S})$, then v is an **External Helly vertex**.

Definition 5.15. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph and IL a feasible vertices insertion list. A vertex $v \in V$ is a **Semi External** Helly vertex, if v is a Helly vertex used by IL, there is at least one pair $(v, S) \in IL$, and v belongs to at most one cluster in $\mathcal{S}$, that is $|\{S \mid v \in S, S \in \mathcal{S}\}| \leq 1$. Note that any external Helly vertex is also a semi external Helly vertex.

Example 5.16. Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph with clusters $\{S_{0,0}, S_{0,1}, S_{0,2}, S_{1,0}, S_{1,2}\}$, where $S_{0,0} = \{1,2\}$, $S_{0,1} = \{2,3\}$, $S_{0,2} = \{3,4\}$, $S_{1,0} = \{1,5\}$, $S_{1,2} = \{4,5,6\}$. The intersection graph $G_{int}(\mathcal{S})$ is described in Figure 22. The vertices in each cluster are specified next to each corresponding node in the intersection graph.

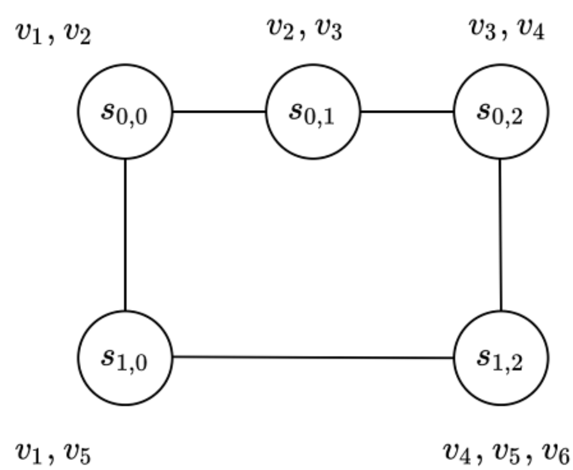


Figure 22: The intersection graph $G_{int}(\mathcal{S})$ of example 5.16

Consider a vertices insertion list IL_1 that includes the pair $(v_2, S_{1,0})$. The intersection graph $G_{int}(\mathcal{S} + IL_1)$ is described in Figure 23 and v_2 is an inner Helly vertx. Note that the red vertices are new vertices inserted to clusters by IL_1.

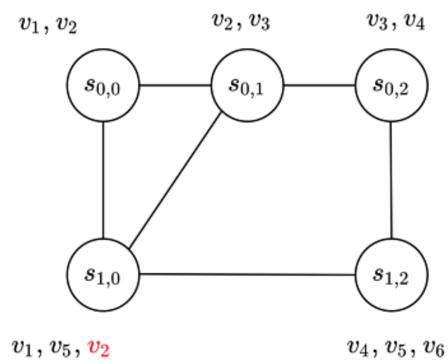


Figure 23: The intersection graph $G_{int}(\mathcal{S} + IL_1)$

Consider a vertices insertion list IL_2 that includes the pairs $\{(v_7, S_{0,2}), (v_7, S_{1,0}), (v_7, S_{1,2})\}$. The intersection graph $G_{int}(\mathcal{S} + IL_2)$ is described in Figure 24 and v_7 is an external Helly vertx. Note that the red vertices are new vertices inserted to clusters by IL_2.

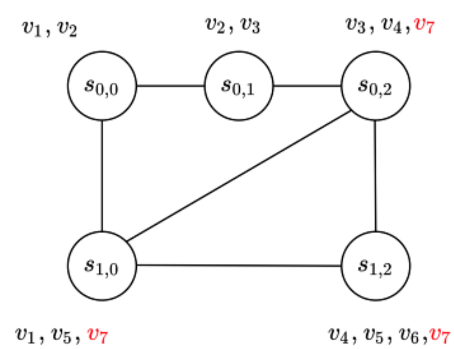


Figure 24: The intersection graph $G_{int}(\mathcal{S} + IL_2)$

Consider a vertices insertion list IL_3 that includes the pairs $\{(v_6, S_{0,0}), (v_6, S_{0,1}), (v_6, S_{1,0})\}$. The intersection graph $G_{int}(\mathcal{S} + IL_3)$ is described in Figure 25 and v_6 is a semi external Helly vertx. Note that the red vertices are new vertices inserted to clusters by IL_3.

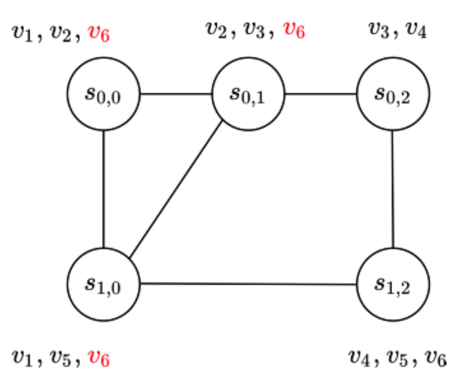


Figure 25: The intersection graph $G_{int}(\mathcal{S} + IL_3)$

Lemma 5.17. Let $H = \langle V, \mathcal{S} \rangle$ be a 1×1 grid hypergraph, if IL is a feasible vertices insertion list with at least one external Helly vertex, then $|IL| \geq 4$.

Proof. The intersection graph $G_{int}(\mathcal{S})$ contains one simple cycle which must be divided into two triangles Δ_1 and Δ_2 using a chord. Without loss of generality, suppose that v^o is an external Helly vertex which corresponds to triangle Δ_1. Since this is an external Helly vertex, it is inserted to all 3 clusters which correspond to the nodes of Δ_1 and thus creates 3 insertions. Consider triangle Δ_2 and its corresponding Helly vertex v_2. Denote by $S' \in \mathcal{S}$ the cluster which correspond to the node in $\Delta_2 \setminus \Delta_1$. Either v_2 is inserted to S' or $v_2 \in S'$.

- If v_2 is inserted into S', then IL contains at least one pair with vertex v_2.
- If $v_2 \in S'$, by the structure of H, v_2 may belong to at most one of the clusters that correspond to nodes in $\Delta_1 \cap \Delta_2$. Since it is the Helly vertex of Δ_2, it is inserted to either one or two of the clusters which correspond to nodes in $\Delta_1 \cap \Delta_2$. Hence, IL also contains at least one pair with vertex v_2.

In any case, IL contains at least 4 insertions, 3 insertion of v^o into the clusters which correspond to the nodes of Δ_1 and at least one insertion with vertex v_2. □

Lemma 5.18. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph, if IL is a feasible vertices insertion list with at least one external Helly vertex, then $|IL| \geq 2n + 2$.

Proof. The proof is by induction of n. According to Lemma 4.4, we assume that for every pair $(v, S) \in IL$, v is a Helly vertex. For $n = 1$, according to Lemma 5.17, $|IL| \geq 2n + 2 = 4$. Suppose the claim of the Lemma is true for $k < n$ and we prove it for n. Suppose IL contains at least one external Helly vertex, and let C be the simple cycle which is highest (of lowest index) in $G_{int}(\mathcal{S})$, such that an external vertex v^o corresponds to one of the triangles of C. Let Δ_1, Δ_2 be the two triangles of C constructed by IL. v^o is an external Helly vertex of Δ_1 which is one of the triangles of C. Let Δ_2 be the other triangle in C, and let v_2 be its Helly vertex.

- Suppose C is the highest cycle in $G_{int}(\mathcal{S})$, containing $s_{0,0}, s_{1,0}, s_{0,1}$ and $s_{1,1}$. Let $H_{down} = H[\mathcal{S} \setminus \{S_{0,0}, S_{0,1}\}]$, H_{down} is an $(n-1) \times 1$ grid hypergraph. According to Lemma 4.2, $IL[H_{down}]$ is a feasible vertices insertion list of H_{down}. Figure 26 shows $G_{int}(\mathcal{S})$ with cycle C the highest.

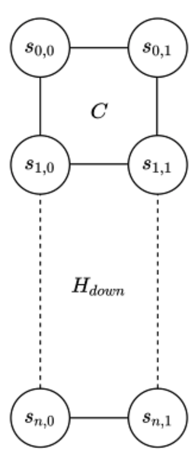


Figure 26: $G_{int}(\mathcal{S})$ with cycle C the highest

- Suppose $IL[H_{down}]$ contains at least one external Helly vertex. According to the induction hypothesis, $|IL[H_{down}]| \geq 2(n-1)+2 = 2n$.
  * Suppose Δ_1 contains nodes $s_{1,0}$ and $s_{1,1}$. In this case, it contains either $s_{0,0}$ or $s_{0,1}$. Without loss of generality, suppose that Δ_1 contains $s_{0,0}$. In that case, $(v^o, S_{0,0}) \in IL \setminus IL[H_{down}]$. In addition, Δ_2 contains $s_{0,1}$. If $v_2 \in S_{0,1}$, then it is inserted to at least one of the clusters $S_{0,0}$ or $S_{1,1}$. If $v_2 \notin S_{0,1}$, then v_2 is inserted into this cluster. In any case, $|IL \setminus IL[H_{down}]| \geq 2$ and $|IL| \geq 2n+2$.
  * Suppose Δ_1 contains exactly one of the nodes $s_{1,0}$ or $s_{1,1}$. In this case, Δ_1 contains both $s_{0,0}$ and $s_{0,1}$ and $\{(v^o, S_{0,0}), (v^o, S_{0,1})\} \subset IL \setminus IL[H_{down}]$. Therefore, $|IL \setminus IL[H_{down}]| \geq 2$ and $|IL| \geq 2n+2$.
- Suppose $IL[H_{down}]$ does not contain any Helly vertex which is external to H_{down}. According to Theorem 5.11, $|IL[H_{down}]| \geq mIL(H_{down}) = 2(n-1) = 2n-2$. v^o is inserted into three clusters corresponding to the three nodes of Δ_1. Let S' be the cluster which corresponds to the node in $\Delta_2 \setminus \Delta_1$. Let $S'' = S' \cup \Big((IL[H_{down}])[S']\Big)$ (a cluster containing all vertices of S' and the vertices inserted to it by $IL[H_{down}]$). Either $v_2 \in S''$ or $(v_2, S') \in IL \setminus IL[H_{down}]$. If $v_2 \in S''$ then it is inserted to at least one of the clusters which correspond to the nodes in $\Delta_2 \cap \Delta_1$. In any case, $IL \setminus IL[H_{down}]$ contains at least one pair with vertex v_2. Hence, $|IL \setminus IL[H_{down}| \geq 4$ and $|IL| \geq 2n+2$.

- C contains $s_{k-1,0}, s_{k,0}, s_{k-1,1}$ and $s_{k,1}$, for $k > 1$. Let $H_{up} = H[S_{0,0}, \ldots, S_{k-1,0}, S_{1,0}, \ldots, S_{k-1,1}]$ and $H_{down} = H[S_{k,0}, \ldots, S_{n,0}, S_{k,1}, \ldots, S_{n,1}]$, H_{up} is a $(k-1)$x1 grid hypergraph and H_{down} is an $(n-k)$x1 grid hypergraph. Note that, since C is the highest cycle to contain an external Helly vertex, v^o is not inserted into any of the clusters in H_{up}. However, if a vertex from the clusters in H_{down} is inserted to any of the clusters in H_{up}, it is an external Helly vertex with respect to H_{up}. In addition, $IL[H_{down}]$ and $IL[H_{up}]$ are disjoint and contained in IL. According to Lemma 4.2, $IL[H_{down}]$ and $IL[H_{up}]$ are feasible vertices insertion lists for H_{down} and H_{up}, respectively. According to Theorem 5.11, $IL[H_{up}] \geq mIL[H_{up}] = 2(k-1)$, $IL[H_{down}] \geq mIL[H_{down}] = 2(n-k)$. Figure 27 shows $G_{int}(\mathcal{S})$ with cycle C in the middle.

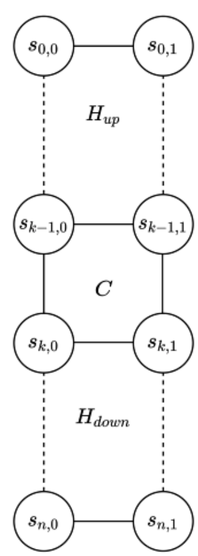


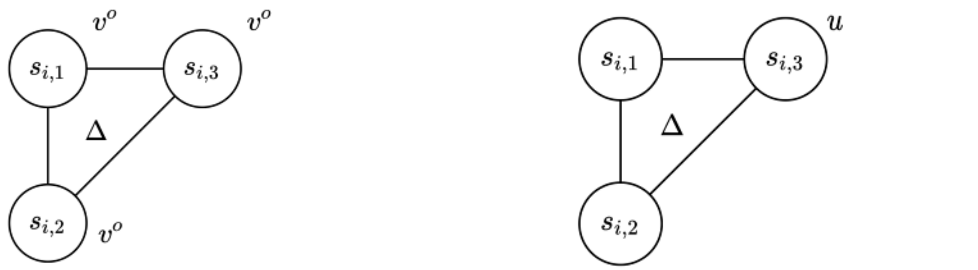
Figure 27: $G_{int}(\mathcal{S})$ with cycle C in the middle

- Suppose $IL[H_{up}]$ does not contain any Helly vertex which is external with respect to H_{up}.
  * Suppose $IL[H_{down}]$ does not contain any Helly vertex which is external with respect to H_{down}. In this case, neither $IL[H_{up}]$ nor $IL[H_{down}]$ contains a pair whose vertex is v^o. Since v^o is the Helly vertex which corresponds to triangle Δ_1, it is inserted to all three clusters that correspond to the nodes of Δ_1 and these three insertions are in $IL \setminus (IL[H_{down}] \cup IL[H_{up}])$. In addition, there is a node of C in $\Delta_2 \setminus \Delta_1$. Without loss of generality, suppose this node is $s_{k,0}$. Let $S'' = S_{k,0} \cup \Big((IL[H_{up}])[S_{k,0}]\Big)$. Either $v_2 \in S''$ or $(v_2, S_{k,0}) \in IL \setminus (IL[H_{down}] \cup IL[H_{up}])$. If $v_2 \in S''$, then it is inserted to at least one of the clusters that correspond to the nodes in $\Delta_2 \cap \Delta_1$. In any case, $|IL \setminus (IL[H_{down}] \cup IL[H_{up}])| \geq 4$ and $|IL| \geq |IL[H_{down}]| + |IL[H_{up}])| + 4 = 2(k-1) + 2(n-k) + 4 = 2n + 2$.
  * Suppose $IL[H_{down}]$ contains at least one Helly vertex which is external with respect to H_{down}. Therefore, according to the induction hypothesis, $|IL[H_{down}]| \geq 2(n-k) + 2$.
    · Suppose v^o is an external Helly vertex in H_{down}. According to the induction hypothesis, $|IL[H_{down}]| \geq 2(n-k) + 2$. Since v^o is the Helly vertex which corresponds to triangle Δ_1, it is inserted to all three clusters that correspond to the nodes of Δ_1. Since at most two of the clusters in Δ_1 are in H_{down}, there is at least one cluster $S \notin H_{down}$ which corresponds to a node $s \in \Delta_1$. $(v^o, S) \notin IL[H_{down}]$ and so $(v^o, S) \in IL \setminus (IL[H_{down}] \cup IL[H_{up}])$. In addition, there is a node of C in $\Delta_2 \setminus \Delta_1$. Without loss of generality, suppose this node is $s_{k,0}$. Let $S'' = S_{k,0} \cup \Big((IL[H_{down}])[S_{k,0}]\Big)$. Either $v_2 \in S''$ or $(v_2, S_{k,0}) \in IL \setminus (IL[H_{down} \cup IL[H_{up}])$. If $v_2 \in S''$, then it is inserted to at least one of the clusters that correspond to the nodes in $\Delta_1 \cap \Delta_2$. In any case, $|IL \setminus (IL[H_{down}] \cup IL[H_{up}])| \geq 2$, and $|IL| \geq |IL[H_{down}]| + |IL[H_{up}]| + 2 = 2(n-k) + 2 + 2(k-1) + 2 = 2n + 2$.
    · Suppose v^o is not an external Helly vertex in H_{down}. Since v^o is the Helly vertex which corresponds to triangle Δ_1, it is inserted to all three clusters that correspond to the nodes of Δ_1 and these three insertions are in $IL \setminus (IL[H_{down}] \cup IL[H_{up}])$. In any case, $|IL \setminus (IL[H_{down}] \cup IL[H_{up}])| \geq 3$, and $|IL| \geq |IL[H_{down}]| + |IL[H_{up}]| + 3 = 2(n-k) + 2 + 2(k-1) + 3 = 2n + 3 \geq 2n + 2$.
- Suppose $IL[H_{up}]$ contains at least one Helly vertex which is external with respect to H_{up}. According to the induction hypothesis, $|IL[H_{up}]| \geq 2(k-1) + 2 = 2k$.
  * Suppose $IL[H_{down}]$ does not contain any Helly vertex which is external with respect to H_{down}. Therefore v^o is not inserted into any of the clusters in H_{down}. Since v^o in an external Helly vertex it is inserted into the three clusters of Δ_1 and these insertions are in $IL \setminus (IL[H_{down}] \cup IL[H_{up}])$. Therefore $|IL| \geq |IL[H_{up}]| + |IL[H_{down}]| + 3 = 2k + 2(n-k) + 3 \geq 2n + 2$.
  * Suppose $IL[H_{down}]$ contains at least one Helly vertex which is external with respect to H_{down}. According to the induction hypothesis, $|IL[H_{up}]| \geq 2(n-k) + 2 = 2n - 2k + 2$. Since $IL[H_{down}]$ and $IL[H_{up}]$ are disjoint and contained in IL, $|IL| \geq |IL[H_{up}]| + |IL[H_{down}]| = 2k + 2n - 2k + 2 = 2n + 2$.

$\square$

Theorem 5.19. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph, and let IL be a feasible vertices insertion list. If IL uses k, $k \geq 0$, different external Helly vertices, then $|IL| \geq 2n + 2k$.

Proof. The proof of the theorem is similar to the proof of Lemma 5.18 and thus we omit the whole proof. However, we will present a different approach to prove this theorem. Consider a triangle Δ, whose Helly vertex v is an external vertex. Let $s_{i_1}, s_{i_2}, s_{i_3}$ be the three nodes of Δ. Adding v to all the corresponding clusters requires three insertions, see Figure 28a. If we consider $u \in S_{i_1} \cap S_{i_2}$ instead, it would require only one insertion, see Figure 28b, so we can perform $IL_{v \to u}$ that leads to a feasible vertices insertion list, such that $|IL_{v \to u}| \leq |IL| - 2$. Continue in this manner until IL', a vertices insertion list with no external Helly vertices, is achieved. $|IL'| \leq |IL| - 2k$. According to Theorem 5.11, $|IL'| \geq 2n$ and therefore, $|IL| \geq 2n + 2k$.



(a) Δ with an external Helly vertex v^o (b) Δ with vertex Helly vertex u instead of v^o

Figure 28: Δ with two options for a Helly vertex

□

Lemma 5.20. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph, and let IL be a feasible vertices insertion list. If IL contains k semi external Helly vertices, then $|IL| \geq 2n + k$.

Proof. The proof of this lemma is similar to the proof of theorem 5.19. Suppose that v is a semi-external Helly vertex used by IL, and let S^* be the only cluster which contains v. Without loss of generality, we can assume that there exists a triangle Δ, such that v is its corresponding Helly vertex and that s^* (the node corresponding S^*) is one of the nodes of Δ (otherwise v is equivalent to an external Helly vertex). Let s^{**} be another node in Δ and S^{**} be its corresponding cluster. In this case, we choose $u \in S^* \cap S^{**}$. Inserting u instead of v reduces the number of insertions by one. Thus, $|IL_{v \to u}| \leq |IL| - 1$, and after changing all the semi external Helly vertices we reach a feasible vertices insertion list IL' with $|IL'| \leq |IL| - k$. □

The following discussion is regarding H a 2×2 grid hypergraph, whose intersection graph is described in Figure 29. For this case we denote $\mathcal{S}_1 = \{S_{0,0}, S_{1,0}, S_{2,0}, S_{0,1}, S_{1,1}, S_{2,1}\}$, $\mathcal{S}_2 = \{S_{0,1}, S_{1,1}, S_{2,1}, S_{0,2}, S_{1,2}, S_{2,2}\}$

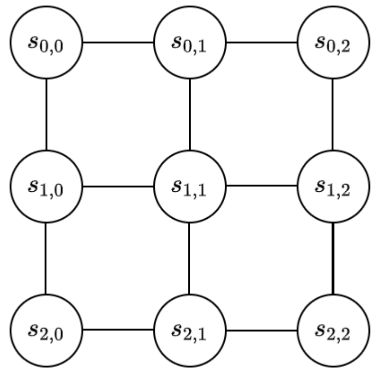


Figure 29: A 2×2 grid hypergraph

Lemma 5.21. Let $H = \langle V, \mathcal{S} \rangle$ be a 2×2 grid hypergraph. If IL is a feasible vertices insertion list of H with $|IL| = 6$, then all Helly vertices are inner Helly vertices.

Proof. Let $H_1 = H[\mathcal{S}_1]$ and $H_2 = H[\mathcal{S}_2]$ be the two induced hypergraphs of H. H_1 and H_2 are both a 2×1 grid hypergraphs. Let $IL_1 = IL[H_1]$, $IL_2 = IL[H_2]$. According to Lemma 4.2, IL_1 and IL_2 are feasible vertices insertion lists for H_1 and H_2 respectively.

Suppose by contradiction that IL contains a pair (v^o, S), where v^o is an external Helly vertex. Without loss of generality, suppose that $(v^o, S) \in IL_1$. According to Lemma 5.18, $|IL_1| \geq 2*2+2 = 6$. According to the assumption of the Lemma, $|IL| = 6$, and therefore $IL = IL_1$ and $IL_2 \setminus IL_1 = \emptyset$. In this case, IL does not contain a pair (v^o, S) with $S \in \{S_{0,2}, S_{1,2}, S_{2,2}\}$.

Suppose $IL = IL_1$ contains (v_1, S_1) and (v_2, S_2) with $v_1 \neq v_2$ and $v_1, v_2 \in (S_{0,2} \cup S_{1,2} \cup S_{2,2}) \setminus V(\mathcal{S}_1)$. According to Lemma 4.4, v_1 and v_2 are both Helly vertices, thus each one of them is inserted to at least three clusters in IL_1. This contradicts the fact that $|IL_1| = 6$ and $(v^o, S) \in IL_1$.

Therefore, there is at most one vertex in $(S_{0,2} \cup S_{1,2} \cup S_{2,2}) \setminus V(\mathcal{S}_1)$ which is contained in $IL_1 = IL$.

Since each vertex is contained in at most two clusters, there is at least one cluster $S^* \in \{S_{0,2}, S_{1,2}, S_{2,2}\}$ such that for every pair $(v', S') \in IL$, $v \notin S^*$ and $S' \neq S^*$, contradicting Lemma 5.6. $\square$

Lemma 5.22. Let $H = \langle V, \mathcal{S} \rangle$ be a 2×2 grid hypergraph. There is no feasible vertices insertion list with cardinality six.

Proof. Let $H_1 = H[\mathcal{S}_1]$ and $H_2 = H[\mathcal{S}_2]$ be the two induced hypergraphs of H. H_1 and H_2 are both a 2×1 grid hypergraphs. Let $IL_1 = IL[H_1], IL_2 = IL[H_2]$. According to Lemma 4.2, IL_1 and IL_2 are feasible vertices insertion lists for H_1 and H_2, respectively. According to Theorem 5.11, $IL_1 \geq 4$, $IL_2 \geq 4$.

Suppose by contradiction that there exists a feasible vertices insertion list IL, whose cardinality is six. Since $IL = IL_1 \cup IL_2$, $|IL_1 \cap IL_2| = |IL_1| + |IL_2| - |IL_1 \cup IL_2| \geq 4 + 4 - 6 = 2$. Hence, $IL_1 \cap IL_2$ contains $\{(v_1, S'), (v_2, S'')\}$, with $S', S'' \in \{S_{0,1}, S_{1,1}, S_{2,1}\} = \mathcal{S}_1 \cap \mathcal{S}_2$. Therefore, $|IL_1 \setminus IL_2| + |IL_2 \setminus IL_1| = 4$.

According to Lemma 5.6, for every cluster $S^* \in (\mathcal{S}_1 \setminus \mathcal{S}_2) \cup (\mathcal{S}_2 \setminus \mathcal{S}_1)$, IL contains a pair (v, S) with either $v \in S^*$ or $S = S^*$. There is a finite small number of options with four insertions to construct the vertices insertion list. For example, there are $S^*, S^{**} \in (\mathcal{S}_1 \setminus \mathcal{S}_2) \cup (\mathcal{S}_2 \setminus \mathcal{S}_1)$ and $v^* \in S^* \cap S^{**}$

such that $IL \setminus (IL_1 \cap IL_2) = \{(v^*, S) \mid S \neq S^*, S \neq S^{**}\}$. However, this is not a feasible vertices insertion list. A proof for the general case is presented in Lemma 5.40. $\square$

Corollary 5.23. Let H be a 2×2 grid hypergraph. If IL is a feasible vertices insertion list, then $|IL| \geq 7$.

Theorem 5.24. Let $H = \langle V, \mathcal{S} \rangle$ be an 2×2 grid hypergraph. There exists IL which is feasible for H such that $|IL| = 7$.

Proof. According to Theorem 4.9, the Convert to Clique method yields a vertices insertion list which is feasible and its cardinality is $|\mathcal{S}| - 2 = 7$. $\square$

Remark 5.25. Let H be a 2×2 grid hypergraph, $H_1 = H[\mathcal{S}_1], H_2 = H[\mathcal{S}_2]$. Note that, if we choose in Convert to Clique method a vertex $v \in S_{0,1} \cap S_{1,1}$. Adding this vertex to $S_{0,0}, S_{1,0}, S_{2,0}, S_{2,1}$ is equivalent to use Convert to Clique method on H_1. Similarly, adding v to $S_{0,2}, S_{1,2}, S_{2,2}, S_{2,1}$ is equivalent to use Convert to Clique method on H_2.

Theorem 5.26. Let $H = \langle V, \mathcal{S} \rangle$ be a 2×2 grid hypergraph. $mIL(H) = 7$.

Proof. According to Corollary 5.23, $mIL(H) \geq 7$. Therefore, according to Theorem 5.24, $mIL(H) = 7$. Hence, $mIL(H) = 7$. $\square$

5.2.3 Outside Insertions

In this section we present special kinds of insertions, side to side insertions and outside insertions. These special kinds of insertions will be used to prove minimum cardinality feasible vertices insertion lists.

Notation 5.27. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph, and IL a feasible vertices insertion list. We denote $\mathcal{S}_l = \{S_{0,0}, \ldots, S_{n,0}\}, V(\mathcal{S}_l) = S_{0,0} \cup \ldots \cup S_{n,0}$, and $\mathcal{S}_r = \{S_{0,1}, \ldots, S_{n,1}\}, V(\mathcal{S}_r) = S_{0,1} \cup \ldots \cup S_{n,1}$, see Figure 30.

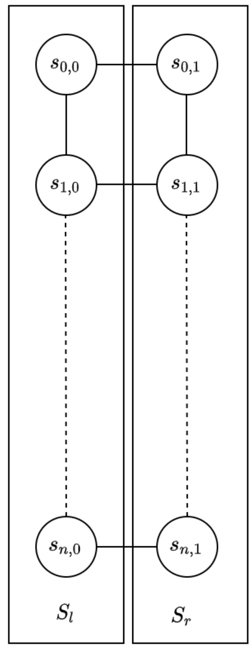


Figure 30: An $n \times 1$ grid hypergraph divided to $\mathcal{S}_l$ and $\mathcal{S}_r$

Definition 5.28. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph, and IL a feasible vertices insertion list. A pair $(v, S) \in IL$ with either $v \in \mathcal{S}_l$ and $S \in \mathcal{S}_r$ or $v \in \mathcal{S}_r$ and $S \in \mathcal{S}_l$ is a **Side to Side insertion**.

Definition 5.29. A pair $(v, S) \in IL$ is an **Outside Insertion with respect to $\mathcal{S}_l$** if exactly one of the following is satisfied:

- $v \in V(\mathcal{S}_l)$ and $S \in \mathcal{S}_r$.
- $v \in V(\mathcal{S}_r)$ and $S \in \mathcal{S}_l$.
- v is an external vertex and $S \in \mathcal{S}_l$.

Similarly, we define **Outside Insertion with respect to $\mathcal{S}_r$**.

Remark 5.30. Clearly, every side to side insertion is both an outside insertion with respect to $\mathcal{S}_l$ and an outside insertion with respect to $\mathcal{S}_r$.

Definition 5.31. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph, and IL a feasible vertices insertion list. $OI(\mathcal{S}_i, IL) = |\{(v, S) \,|\, (v, S) \in IL, (v, S)$ is an outside insertion with respect to $\mathcal{S}_i$ (where $\mathcal{S}_i = \mathcal{S}_l$ or $\mathcal{S}_i = \mathcal{S}_r)\}|$, the **Number of Outside Insertions with respect to $\mathcal{S}_i$** in IL.

Lemma 5.32. Let $H = \langle V, \mathcal{S} \rangle$ be a 1×1 grid hypergraph. If IL is a feasible vertices insertion list, then IL contains at least two side to side insertions.

Proof. Consider $G_{int}(\mathcal{S})$ which is 1×1 grid hypergraph, as described in Figure 31a. $\mathcal{S}_l = \{S_{0,0}, S_{1,0}\}$, and $\mathcal{S}_r = \{S_{0,1}, S_{1,1}\}$.

(a) A 1×1 grid hypergraph

(b) A 1×1 grid hypergraph with a chord $s_{1,0}, s_{0,1}$

Figure 31: A 1×1 grid hypergraph before and after adding a chord

Since IL is a feasible vertices insertion list in $G_{int}(\mathcal{S} + IL)$, the cycle contains at least one chord, which divides the cycle into two triangles, Δ_1 and Δ_2. One option is $\Delta_1 = \{S_{0,0}, S_{1,0}, S_{0,1}\}$ and $\Delta_2 = \{S_{1,0}, S_{0,1}, S_{1,1}\}$. We will prove the lemma for this option, see Figure 31b. According to Theorem 3.2, since IL is a feasible vertices insertion list, $H + IL$ satisfies the Helly property. Therefore, the clusters which correspond to the nodes in Δ_1 share a common vertex. Hence, one of the following holds:

- IL contains pair $(v, S_{1,0}), v \in S_{0,0} \cap S_{0,1}$, which is a side to side insertion since $v \in S_{0,1} \subseteq V(\mathcal{S}_r)$ and $S_{1,0} \in \mathcal{S}_l$.
- IL contains pair $(v, S_{0,1}), v \in S_{0,0} \cap S_{1,0}$, which is a side to side insertion since $v \in S_{1,0} \subseteq V(\mathcal{S}_l)$ and $S_{0,1} \in \mathcal{S}_r$.

Similarly for Δ_2, one of the following holds:

- IL contains pair $(v, S_{1,0}), v \in S_{0,1} \cap S_{1,1}$, which is a side to side insertion since $v \in S_{0,1} \subseteq V(\mathcal{S}_r)$ and $S_{1,0} \in \mathcal{S}_l$.
- IL contains pair $(v, S_{0,1}), v \in S_{1,0} \cap S_{1,1}$, which is a side to side insertion since $v \in S_{1,0} \subseteq V(\mathcal{S}_l)$ and $S_{0,1} \in \mathcal{S}_r$.

In any case, each triangle corresponds to at least one side to side insertion, and IL contains two side to side insertions. Another option is $\Delta_1 = \{S_{0,0}, S_{1,0}, S_{1,1}\}$ and $\Delta_2 = \{S_{0,0}, S_{0,1}, S_{1,1}\}$. A similar proof holds for this option. $\square$

Remark 5.33. We note that IL may also be composed using the Convert to Clique method. In this case we choose a vertex from the intersection of two clusters and insert it to the other two clusters in $\mathcal{S}$. $G_{int}(\mathcal{S} + IL)$ is demonstrated in Figure 32.

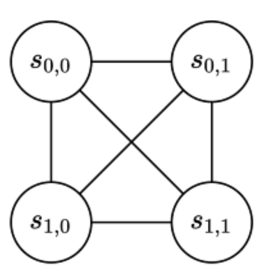


Figure 32: A 1×1 grid hypergraph after using Convert to Clique method

Suppose $v \in S_{0,0} \cap S_{1,0}, v \in V(\mathcal{S}_l)$. In this case, IL contains pairs $(v, S_{0,1}), (v, S_{1,1})$ where $S_{0,1}, S_{1,1} \in V(\mathcal{S}_r)$. Suppose $v \in S_{0,1} \cap S_{1,1}, v \in V(\mathcal{S}_r)$. In this case, IL contains pairs $(v, S_{0,0}), (v, S_{1,0})$, where $S_{0,0}, S_{1,0} \in V(\mathcal{S}_l)$. Suppose $v \in S_{0,0} \cap S_{0,1}, v \in V(\mathcal{S}_r), v \in V(\mathcal{S}_l)$. In this case, IL contains pairs $(v, S_{1,0}), (v, S_{1,1})$ where $S_{1,0} \in V(\mathcal{S}_l), S_{1,0} \in V(\mathcal{S}_r)$. $(v, S_{1,0})$ is a side to side insertion since $v \in V(\mathcal{S}_r)$ and $S_{1,0} \in V(\mathcal{S}_l)$. $(v, S_{1,1})$ is a side to side insertion since $v \in V(\mathcal{S}_l)$ and $S_{1,1} \in V(\mathcal{S}_r)$. A similar proof holds for $v \in S_{1,0} \cap S_{1,1}$. Therefore, in any case IL contains two side to side insertions.

Lemma 5.34. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph. Let IL be a feasible vertices insertion list of H, then for every basic triangle Δ, IL contains at least one outside insertion with respect to $\mathcal{S}_i$ (for $\mathcal{S}_i = \mathcal{S}_l$ or $\mathcal{S}_i = \mathcal{S}_r$), which contains the Helly vertex corresponding to Δ.

Proof. Suppose, without loss of generality, Δ contains two nodes in $\mathcal{S}_l$, and one in $\mathcal{S}_r$, see Figure 33.

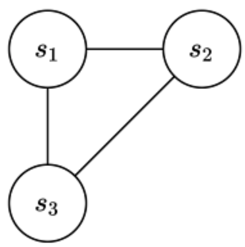


Figure 33: A triangle with two nodes in $\mathcal{S}_l$ and one node in $\mathcal{S}_r$

There are four options for v:

- $v \in V(\mathcal{S}_l) \setminus V(\mathcal{S}_r)$. In this case, $v \in S$ for $S \in \mathcal{S}_l$. Since v is a Helly vertex of Δ, v was inserted into $S_2 \in \mathcal{S}_r$. In this case, IL contains pair (v, S_2) which is a side to side insertion.
- $v \in V(\mathcal{S}_r) \setminus V(\mathcal{S}_l)$. In this case, $v \in S$ for $S \in \mathcal{S}_r$. Since v is a Helly vertex of Δ, v was inserted into $S_1, S_3 \in \mathcal{S}_l$. In this case, IL contains pairs $(v, S_1), (v, S_3)$ which are each a side to side insertion.
- $v \in S_i \cap S_j$ for $S_i \in \mathcal{S}_l$ and $S_j \in \mathcal{S}_r$. In this case, $v \in V(\mathcal{S}_l)$ and $v \in V(\mathcal{S}_r)$. Since a vertex is contained in at most two clusters in H, there exists a cluster S in Δ such that IL contains pair (v, S). if $S \in \mathcal{S}_l$, then (v, S) is a side to side insertion since $v \in V(\mathcal{S}_r)$. if $S \in \mathcal{S}_r$, then (v, S) is a side to side insertion since $v \in V(\mathcal{S}_l)$.
- v is an external Helly vertex. In this case, v is inserted to S_1 and S_2. (v, S_1) is an outside insertion with respect to $\mathcal{S}_l$ and (v, S_2) is an outside insertion with respect to $\mathcal{S}_r$.

In all four cases Δ contains at least one outside insertion. □

Definition 5.35. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph and let IL be a feasible vertices insertion list. Let Δ_1 and Δ_2 be two basic triangles in $G_{int}(\mathcal{S} + IL)$ which share an edge. Δ_1 and Δ_2 form one of the following shapes:

1. A **cycle** as described in Figure 34.
2. A **diamond** as described in Figure 35.

3. A **flag** as described in Figure 36.

(a) A basic cycle with two basic triangles that share an edge - option 1

(b) A basic cycle with two basic triangles that share an edge - option 2

Figure 34: Two adjacent triangles in a basic cycle

(a) A diamond shape with two basic triangles that share an edge - option 1

(b) A diamond shape with two basic triangles that share an edge - option 2

Figure 35: A diamond shape with triangles

(a) A flag shape with two basic triangles that share an edge - option 1

(b) A flag shape with two basic triangles that share an edge - option 2

Figure 36: A flag shape with triangles

Lemma 5.36. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph and let IL be a feasible vertices insertion list. If Δ_1 and Δ_2 are two basic triangles which share an edge and create a cycle C, then the Helly vertices correspond to at least two different outside insertions with respect to $\mathcal{S}_i$, for $\mathcal{S}_i = \mathcal{S}_l$ or $\mathcal{S}_i = \mathcal{S}_r$.

Proof. There are two options for the way C contains Δ_1 and Δ_2, as described in Figure 34. Let

$C = s_{i,0} - s_{i,1} - s_{i+1,1} - s_{i+1,0} - s_{i,0} \in G_{int}(\mathcal{S})$ be the simple cycle which contains Δ_1 and Δ_2. Since IL is a feasible vertices insertion list, according to Theorem 3.2, $G_{int}(\mathcal{S} + IL)$ is chordal. Therefore, at least one chord is added to C in $G_{int}(\mathcal{S} + IL)$. There are two options for this chord:

- $(s_{i,0}, s_{i+1,1})$ as described in Figure 34a.
- $(s_{i,1}, s_{i+1,0})$ as described in Figure 34b.

Let v_1 and v_2 be the Helly vertices which correspond to Δ_1 and Δ_2, respectively. If $v_1 \neq v_2$, then according to Lemma 5.34, IL contains at least one outside insertion with respect to $\mathcal{S}_i$ which corresponds to v_1 and one outside insertion with respect to $\mathcal{S}_i$ which corresponds to v_2. Since $v_1 \neq v_2$ these are two different outside insertions. Figure 37 shows an example of the two outside insertions where $v_1 \neq v_2$, $(v_1, S_{i+1,1})$ where $v_1 \in S_{i,0} \cap S_{i+1,0}$ and $(v_2, S_{i+1,1})$ where $v_2 \in S_{i,0} \cap S_{i,1}$.

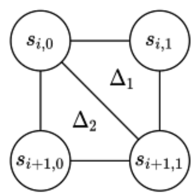


Figure 37: Two adjacent triangles in a basic cycle

If $v_1 = v_2 = v$, without loss of generality, suppose Δ_1 and Δ_2 are arranged according to Figure 37. The shared edge is $(s_{i,0}, s_{i+1,1})$. There are four options for v:

- $v \in V(\mathcal{S}_l) \setminus V(\mathcal{S}_r)$. In this case, IL contains $(v, S_{i,1})$ and $(v, S_{i+1,1})$ which are two different side to side insertions.
- $v \in V(\mathcal{S}_r) \setminus V(\mathcal{S}_l)$. In this case, IL contains $(v, S_{i,0})$ and $(v, S_{i+1,0})$ which are two different side to side insertions.
- $v \in V(\mathcal{S}_l) \cap V(\mathcal{S}_r)$. In this case, $v \in S_j \cap S_k$ for $S_j \in \mathcal{S}_l$ and $S_k \in \mathcal{S}_r$. Hence, either $v \notin S_{i,0} \cup S_{i,1}$ or $v \notin S_{i+1,0} \cup S_{i+1,1}$. If $v \notin S_{i+1,0} \cup S_{i+1,1}$, then IL contains $(v, S_{i+1,0})$, which is a side to side insertion, since $v \in V(\mathcal{S}_r)$ and $S_{i+1,0} \in \mathcal{S}_l$, and IL contains $(v, S_{i+1,1})$, which is a side to side insertion, since $v \in V(\mathcal{S}_l)$ and $S_{i+1,1} \in \mathcal{S}_r$. A similar proof holds for the case $v \notin S_{i,0} \cup S_{i,1}$.
- v is an external Helly vertex. In this case, IL contains $(v, S_{i,0})$ and $(v, S_{i+1,0})$ which are outside insertions with respect to $\mathcal{S}_l$, and IL contains $(v, S_{i,1})$ and $(v, S_{i+1,1})$ which are outside insertions with respect to $\mathcal{S}_r$.

□

Lemma 5.37. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph and let IL be a feasible vertices insertion list. Let Δ_1, Δ_2 be two basic triangles which share an edge and create a diamond, then the Helly vertices which correspond to Δ_1 and Δ_2 create at least two outside insertions with respect to $\mathcal{S}_i$, for $\mathcal{S}_i = \mathcal{S}_l$ or $\mathcal{S}_i = \mathcal{S}_r$.

Proof. Since the two cases of the diamond shape are symmetric, without loss of generality, we discuss the case described in Figure 35a. In this case, $\Delta_1 = \{S_{i,0}, S_{i+1,0}, S_{i+1,1}\}$ and $\Delta_2 = \{S_{i+1,0}, S_{i+1,1}, S_{i+2,1}\}$.

Let v_1 and v_2 be the Helly vertices which correspond to Δ_1 and Δ_2, respectively. If $v_1 \neq v_2$, then according to Lemma 5.34, IL contains at least one outside insertion with respect to $\mathcal{S}_i$ which corresponds to v_1 and one outside insertion with respect to $\mathcal{S}_i$ which corresponds to v_2. Since $v_1 \neq v_2$ these are two different side to side insertions. Figure 38 shows two triangles forming a diamond shape and Δ_1, Δ_2.

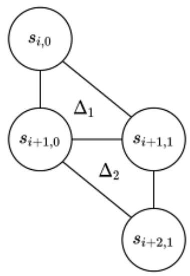


Figure 38: A diamond shape

If $v_1 = v_2 = v$ there are four options for v:

- $v \in V(\mathcal{S}_l) \setminus V(\mathcal{S}_r)$. In this case, IL contains $(v, S_{i+1,j+1})$ and $(v, S_{i+2,j+1})$ which are two different side to side insertions.
- $v \in V(\mathcal{S}_r) \setminus V(\mathcal{S}_l)$. In this case, IL contains $(v, S_{i,j})$ and $(v, S_{i+1,j})$ which are two different side to side insertions.
- $v \in V(\mathcal{S}_l) \cap V(\mathcal{S}_r)$. In this case, $v \in S_j \cap S_k$ for $S_j \in \mathcal{S}_l$ and $S_k \in \mathcal{S}_r$. Hence, either $v \in S_{i,0} \cap S_{i,1}$ or $v \in S_{i+1,0} \cap S_{i+1,1}$ or $v \in S_{i+2,0} \cap S_{i+2,1}$ or $[v \notin S_{i,0} \cup S_{i,1} \cup S_{i+1,0} \cup S_{i+1,1} \cup S_{i+2,0} \cup S_{i+2,1}]$.
  - If $v \in S_{i,0} \cap S_{i,1}$, then IL contains $(v, S_{i+1,0})$, which is a side to side insertion since $v \in V(\mathcal{S}_r)$ and $S_{i+1,0} \in \mathcal{S}_l$, and IL contains $(v, S_{i+1,1})$ which is a side to side insertion since $v \in V(\mathcal{S}_l)$ and $S_{i+1,1} \in \mathcal{S}_r$.
  - If $v \in S_{i+1,0} \cap S_{i+1,1}$, then IL contains $(v, S_{i,0})$, which is a side to side insertion since $v \in V(\mathcal{S}_r)$ and $S_{i,0} \in \mathcal{S}_l$, and IL contains $(v, S_{i+2,1})$ which is a side to side insertion since $v \in V(\mathcal{S}_l)$ and $S_{i+2,1} \in \mathcal{S}_r$.
  - If $v \in S_{i+2,0} \cap S_{i+2,1}$, then IL contains $(v, S_{i+1,0})$, which is a side to side insertion since $v \in V(\mathcal{S}_r)$ and $S_{i+1,0} \in \mathcal{S}_l$, and IL contains $(v, S_{i+1,1})$ which is a side to side insertion since $v \in V(\mathcal{S}_l)$ and $S_{i+1,1} \in \mathcal{S}_r$.
  - If $[v \notin S_{i,0} \cup S_{i,1} \cup S_{i+1,0} \cup S_{i+1,1} \cup S_{i+2,0} \cup S_{i+2,1}]$, then IL contains $(v, S_{i,0})$, which is a side to side insertion since $v \in V(\mathcal{S}_r)$ and $S_{i,0} \in \mathcal{S}_l$, and IL contains $(v, S_{i+1,1})$ which is a side to side insertion since $v \in V(\mathcal{S}_l)$ and $S_{i+1,1} \in \mathcal{S}_r$.
- v is an external Helly vertex. In this case, IL contains pairs $(v, S_{i,0})$ and $(v, S_{i+1,0})$ which are outside insertions with respect to $\mathcal{S}_l$ and $(v, S_{i,1})$ and $(v, S_{i+1,1})$ which are outside insertions with respect to $\mathcal{S}_r$.

□

Remark 5.38. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph and let IL be a feasible vertices insertion list. If Δ_1 and Δ_2 are two basic triangles which share an edge and create a flag, the Helly vertices which correspond to Δ_1 and Δ_2 may correspond to only one outside insertion.

Figure 39 shows the two possible cases. Consider Figure 39a. Let $\Delta_1 = \{s_{i,0}, s_{i+1,0}, s_{i+1,1}\}, \Delta_2 = \{s_{i+1,0}, s_{i+2,0}, s_{i+1,1}\}$. For Δ_1, either a vertex $v \in S_{i+1,0} \cap S_{i+1,1}$ is added to $S_{i,0}$ or a vertex $v \in S_{i,0} \cap S_{i+1,0}$ is added to $S_{i+1,1}$ or a vertex $v \in S_{i,0} \cap S_{i+1,1}$ is added to $S_{i+1,0}$. For Δ_2, either a vertex $v \in S_{i+1,0} \cap S_{i+1,1}$ is added to $S_{i+2,0}$ or a vertex $v \in S_{i+1,0} \cap S_{i+2,0}$ is added to $S_{i+1,1}$ or a vertex $v \in S_{i+2,0} \cap S_{i+1,1}$ is added to $S_{i+1,0}$. In a similar way, only one outside insertion exists in the two triangles in Figure 39b.

(a) A flag shape with two basic triangles that share an edge - option 1

(b) A flag shape with two basic triangles that share an edge - option 2

Figure 39: A flag shape

Theorem 5.39. Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 1$ grid hypergraph and let IL be a feasible vertices insertion list. IL contains at least $n+1$ outside insertions with respect to $\mathcal{S}_i$, for $\mathcal{S}_i = \mathcal{S}_l$ or $\mathcal{S}_i = \mathcal{S}_r$.

Proof. Assume that $G_{int}(\mathcal{S} + IL)$ includes k different Helly vertices, $v_1 \ldots, v_k$. Assume that v_i is a Helly vertex in y_i different basic triangles. Thus,

$$\sum_{i=1}^{k} y_i = 2n$$

According to Lemma 5.34, every Helly vertex creates in each basic triangle at least one outside insertion. A Helly vertex may correspond to two basic triangles which share an edge but only one outside insertion with respect to $\mathcal{S}_i$, for $\mathcal{S}_i = \mathcal{S}_l$ or $\mathcal{S}_i = \mathcal{S}_r$.

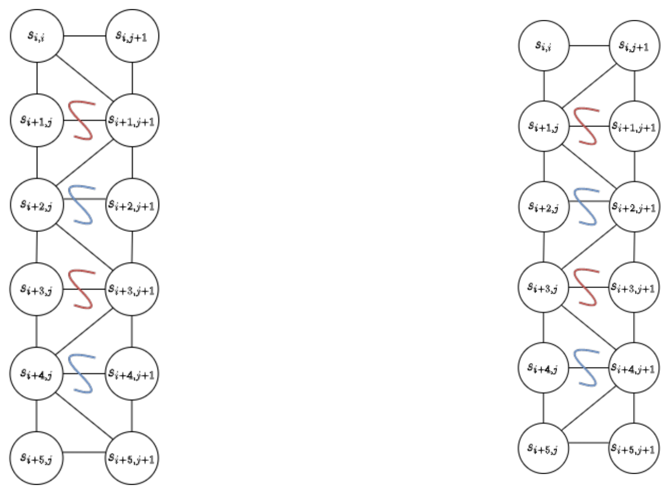
Define

$$x_i = \begin{cases} 1 & \text{If vertex } v_i \text{ is a Helly vertex which corresponds} \\ & \text{to two basic triangles which share an edge and form a flag} \\ 0 & \text{otherwise} \end{cases}$$

Therefore, the number of side to side insertions is at least:

$$\sum_{i=1}^{k} y_i - \sum_{i=1}^{k} x_i = 2n - \sum_{i=1}^{k} x_i$$

The maximum number of pairs of basic triangles that share an edge and create a flag is $\sum_{i=1}^{k} x_i \leq n-1$, since a flag is two basic triangles which belong to two different basic cycles. Figure 40 shows multiple basic triangles creating multiple flag shapes. Note that each curly line highlights a different flag shape.



(a) Multiple basic triangles creating multiple flag shapes - option 1

(b) Multiple basic triangles creating multiple flag shapes - option 2

Figure 40: A flag shape for $n \times 1$ grid hypergraph

Finally, we get that the number of outside insertions with respect to $\mathcal{S}_i$ is at least:

$$\sum_{i=1}^{k} y_i - \sum_{i=1}^{k} x_i \geq 2n - (n-1) = n+1$$

□

We consider now $n \times 2$ grid hypergraphs, using the results of $n \times 1$ grid hypergraphs.

Lemma 5.40. *Let $H = \langle V, \mathcal{S} \rangle$ be an $n \times 2$ grid hypergraph and let IL be a feasible vertices insertion list. $|IL| > 3n$.*

Proof. Suppose by contradiction that $|IL| = 3n$. According to Lemma 4.4, every vertex v such that there is a pair $(v, S) \in IL$ is a Helly vertex. Let $\mathcal{S}_1 = \{S_{0,0}, \ldots, S_{n,0}, S_{0,1}, \ldots, S_{n,1}\}$, $\mathcal{S}_2 = \{S_{0,1}, \ldots, S_{n,1}, S_{0,2}, \ldots, S_{n,2}\}$, $H_1 = H[\mathcal{S}_1], H_2 = H[\mathcal{S}_2]$ and $IL_1 = IL[H_1], IL_2 = IL[H_2]$.

Let $V^{o,2}$ be the set of different vertices from $S_{0,2} \cup \ldots \cup S_{n,2}$ which are inserted by IL to clusters in $\mathcal{S}_1$, $V^{o,2} = \{v \in S_{0,2} \cup \ldots \cup S_{n,2} \,|\, \exists (v, S) \in IL, S \in \mathcal{S}_1\}$. Note that, every vertex in $V^{o,2}$ is an external Helly vertex or semi external vertex in IL_1.

According to Lemma 4.2, IL_1 is a feasible vertices insertion list of H_1. According to Lemma 5.20, $|IL_1| \geq 2n + |V^{o,2}|$. Therefore, $|IL_2 \setminus IL_1| \leq |IL| - |IL_1| = 3n - (2n + |V^{o,2}|) = n - |V^{o,2}|$.

According to Lemma 4.2, IL_2 is a feasible vertices insertion list of H_2, and $IL_2 \setminus IL_1$ contains all the insertions into clusters $S_{0,2} \cup \ldots \cup S_{n,2}$, $IL_2 \setminus IL_1 = \{(v, S) \,|\, (v, S) \in IL_2, S \in S_{0,2} \cup \ldots \cup S_{n,2}\}$.

The number of $OI(\{S_{0,2},\dots,S_{n,2}\},IL_2)$ is bounded by $|IL_2 \setminus IL_1|+|V^{o,2}| \le n-|V^{o,2}|+|V^{o,2}| \le n$. This is a contradiction to Theorem 5.39 which states that $OI(\{S_{0,2},\dots,S_{n,2}\},IL_2) \ge n+1$. □

Theorem 5.41. Let $H=\langle V,\mathcal{S}\rangle$ be an $n\times 2$ grid hypergraph. $mIL(H)=3n+1$.

Proof. According to Lemma 5.40, a vertices insertion list IL whose cardinality is $3n$ cannot be feasible. Therefore, the minimum cardinality of a vertices insertion list is at least $3n+1$. The number of nodes in $G_{int}(\mathcal{S})$ is $3(n+1)$. According to Theorem 4.9, a vertices insertion list created by the Convert to Clique method, is a feasible solution with cardinality $3(n+1)-2=3n+1$. Therefore, $mIL(H)=3n+1$. □

Remark 5.42. Let $H=\langle V,\mathcal{S}\rangle$ be an $n\times m$ grid hypergraph. Let IL be a feasible vertices insertion list of H. Since IL is a feasible vertices insertion list in $G_{int}(\mathcal{S}+IL)$, each basic cycle contains at least one chord, which divides the cycle into two triangles. However, creating x triangles does not necessarily require x insertions. For example, consider an $n\times 2$ grid hypergraph, for $n>1$, and let IL be a feasible vertices insertion list created using the Convert to Clique method achieving $|IL|=3n+1$.

On the other hand, IL creates more than $4n$ triangles in H, and we get $|IL|=3n+1<4n$.

6 Chords Addition

This section contains results regarding feasible and minimum cardinality feasible chords addition list whose addition to a graph achieves chordality.

The following theorem is a known result proved by Levin [9].

Theorem 6.1. Let C be a chordless cycle of size $|C| \geq 3$, then $mAL(C) = |C| - 3$.

Example 6.2. Let $C = s_{0,0} - s_{0,1} - s_{0,2} - s_{1,2} - s_{2,2} - s_{2,1} - s_{2,0} - s_{1,0} - s_{0,0}$ be a cycle of size $|C| = 8$. According to Theorem 6.1, mAL(C) = 5. A possible minimum cardinality feasible chords addition list is: $AL = \{(s_{0,0}, s_{0,2}), (s_{0,0}, s_{1,2}), (s_{0,0}, s_{2,2}), (s_{0,0}, s_{2,1}), (s_{0,0}, s_{2,0})\}$, see Figure 41. The edges of C are colored black and the edges in AL are colored red.

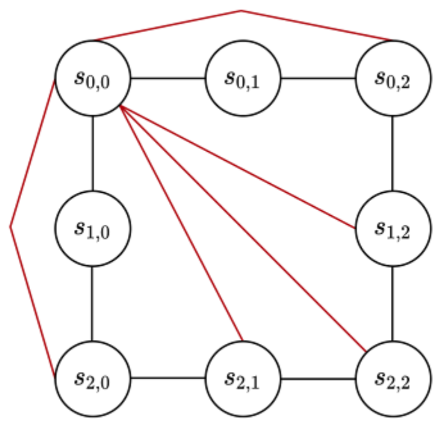
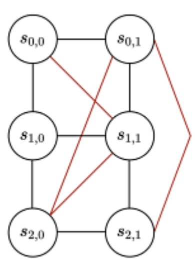
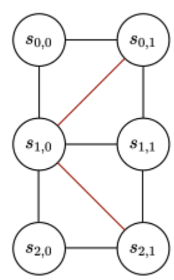


Figure 41: A minimum cardinality feasible chords addition list for a cycle C

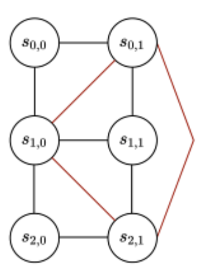
Let $G = \langle U, E \rangle$ be a 2×1 grid graph. The chords addition lists in Figure 42 are feasible.



(a) Option 1



(b) Option 2



(c) Option 3

Figure 42: A 2×1 grid with feasible chords addition lists

However, the chords addition list in Figure 43 is not feasible. The reason AL is not feasible is that $G + AL$ still contains cycle $s_{0,0} - s_{0,1} - s_{2,1} - s_{2,0} - s_{1,0} - s_{0,0}$.

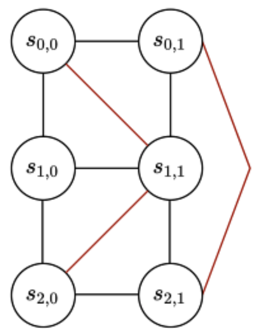


Figure 43: A 2×1 grid with a chords addition list that is not feasible

Definition 6.3. Let $G = \langle U, E \rangle$ be an $n \times m$ grid graph. A **Clipping Corner chord** is one of the following chords:

- $(s_{0,1}, s_{1,0})$.
- $(s_{n-1,0}, s_{n,1})$.
- $(s_{0,m-1}, s_{1,m})$.
- $(s_{n-1,m}, s_{n,m-1})$.

Figure 44 describes a graph with four clipping corners chords.

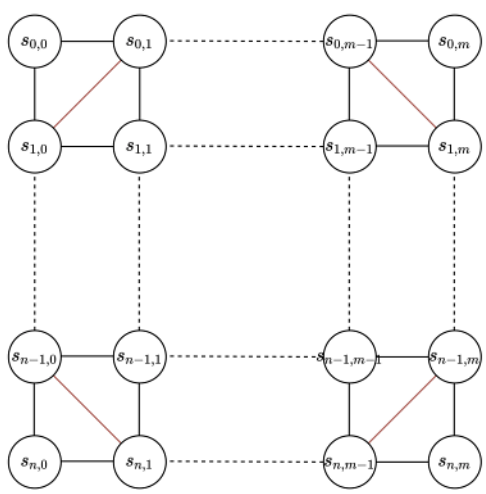


Figure 44: A graph with four clipping corner chords

We will use this definition throughout this section.

In all figures, the edges of G are colored black and the edges in AL are colored red.

6.1 2×2 grid graphs

In this section we present a minimum cardinality feasible chords addition list for 2×2 grid graphs.

Let $G = \langle U, E \rangle$ be a 2×2 grid graph. G contains four basic cycles: $C_1 = s_{0,0} - s_{0,1} - s_{1,1} - s_{1,0} - s_{0,0}$, $C_2 = s_{1,0} - s_{1,1} - s_{2,1} - s_{2,0} - s_{1,0}$, $C_3 = s_{0,1} - s_{0,2} - s_{1,2} - s_{1,1} - s_{0,1}$, $C_4 = s_{1,1} - s_{1,2} - s_{2,2} - s_{2,1} - s_{1,1}$. Furthermore, G contains eight node cycle $C_8 = s_{0,0} - s_{0,1} - s_{0,2} - s_{1,2} - s_{2,2} - s_{2,1} - s_{2,0} - s_{1,0} - s_{0,0}$.

Lemma 6.4. Let $G = \langle U, E \rangle$ be a 2×2 grid graph. Let $AL = \{(s_{0,1}, s_{1,0}), (s_{1,0}, s_{2,1}), (s_{0,1}, s_{2,1}), (s_{0,1}, s_{1,2}), (s_{1,2}, s_{2,1})\}$. AL is a feasible chords addition list of G.

Proof. To shows that AL is a feasible chords addition list we need to show that $G + AL$ is chordal. Figure 45 demonstrates $G + AL$, where the edges of G are colored black and the edges of AL are colored red.

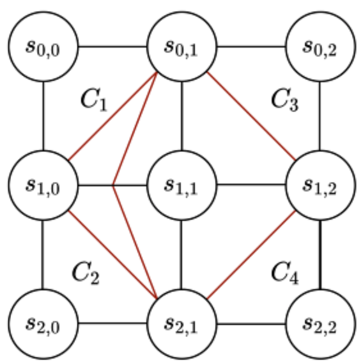


Figure 45: A 2×2 grid graph

Assume by contradiction that $G + AL$ contains C a chordless cycle, $|C| \geq 4$. C could reside in:

- Inside one of the basic cycles. In this case, the size of C is four at most. However, AL creates a chord in each basic cycle.
- Inside two basic cycles. In this case, C cannot reside in (C_1 and C_4) or (C_2 and C_3). The reason is that if C resides in ($C1$ and $C4$) or (C_2 and C_3), it has to pass through $s_{1,1}$ twice, contradicting the fact that C is a simple cycle. Therefore, C can reside in two adjacent cycles, without loss of generality, C_1 and C_2. In this case, the edge shared by the two simple cycles $(s_{1,0}, s_{1,1})$ is a chord of C.
- Inside three or four basic cycles and pass through $s_{1,1}$. In this case, there is a path P which contains at least one of edges $(s_{0,1}, s_{1,1}), (s_{1,1}, s_{2,1}), (s_{1,0}, s_{1,1}), (s_{1,1}, s_{1,2})$. This edge is a chord of C.
- C is cycle $C_8 = s_{0,0} - s_{0,1} - s_{0,2} - s_{1,2} - s_{2,2} - s_{2,1} - s_{2,0} - s_{1,0} - s_{0,0}$. In this case $(s_{0,1}, s_{2,1})$ is a chord of C.

Thus, AL is a feasible chords addition list of G. □

Lemma 6.5. Let $G = \langle U, E \rangle$ be a 2×2 grid graph. $mAL(G) = 5$.

Proof. G contains the following eight node cycle: $C = s_{0,0} - s_{0,1} - s_{0,2} - s_{1,2} - s_{2,2} - s_{2,1} - s_{2,0} - s_{1,0} - s_{0,0}$. According to Theorem 6.1, C requires at least five chords to achieve chordality. According to Lemma 6.4, the chords addition list AL described in the lemma is feasible and $|AL| = 5$. Therefore, $mAL(G) = 5$. □

Lemma 6.6. Let $G = \langle U, E \rangle$ be a 2×2 grid graph. A minimum cardinality feasible chords addition list must contain four clipping corner chords.

Proof. Each one of the basic cycles requires at least one chord to achieve chordality in that cycle. Suppose by contradiction, AL is a minimum cardinality feasible chords addition list which contains at least one chord inside a basic cycle which is not a clipping corner chord, denote this chord e. This chord is one of the following edges: $(s_{0,0}, s_{1,1}), (s_{2,0}, s_{1,1}), (s_{0,2}, s_{1,1})$ or $(s_{2,2}, s_{1,1})$. Any of these chords are not a chord of C_8. According to Theorem 6.1, C_8 requires at least five chords to achieve chordality. Therefore, AL contains e and at least five more chords. Hence, $|AC| \geq 6$ contradicting Lemma 6.5, which states that $mAL(G) = 5$. □

6.2 $n \times 1$ One sided clique grid graphs

In this section we present a special case of graphs, one sided clique grid graph, defined in Definition 6.7. For these graphs we present feasible chords addition lists.

Definition 6.7. A graph $G = \langle U, E \rangle$ with $U = \{s_{0,0}, \ldots, s_{n,0}, s_{0,1}, \ldots, s_{n,1}\}$ is an $n \times 1$ **One Sided Clique Grid** graph if G is composed of exactly $n \times 1$ grid graph and a clique on $s_{0,0}, \ldots, s_{n,0}$, see Figure 46.

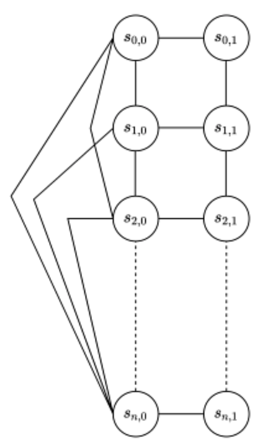


Figure 46: An $n \times 1$ one sided clique grid graph

6.2.1 2×1 One sided clique grid graphs

In this section we present a minimum cardinality feasible chords addition list for 2×1 one sided clique grid graphs.

Consider the 2×1 one sided clique grid graph in Figure 47.

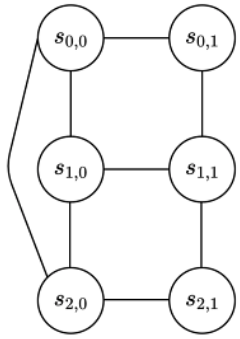


Figure 47: A 2×1 one sided clique grid graph

Lemma 6.8. Let $G = \langle U, E \rangle$ be a 2×1 one sided clique grid graph. The chords addition list $AL = \{(s_{0,0}, s_{1,1}), (s_{1,1}, s_{2,0})\}$ is a feasible chords addition list (see Figure 48).

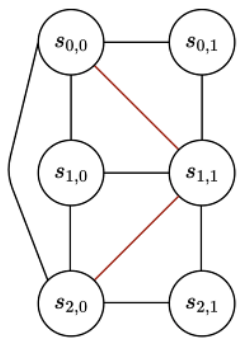


Figure 48: A 2×1 one sided clique grid graph. The edges of G are colored black and the edges in AL are colored red

Proof. G contains the following simple cycles: $C_1 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{2,0} - s_{0,0}$, $C_2 = s_{0,0} - s_{0,1} - s_{1,1} - s_{1,0} - s_{0,0}$, $C_3 = s_{1,0} - s_{1,1} - s_{2,1} - s_{2,0} - s_{1,0}$, see Figure 48. AL adds chords to these cycles, achieving chordality on all these cycles, without creating new cycles. Therefore, $G + AL$ is chordal. $\square$

We will now present another proof for the Lemma which uses the connection between CFC and CST. This proof is based on another approach to check whether an instance for the CST has a feasible solution tree presented in Figure 48.

Definition 6.9. Given a hypergraph $H = \langle V, \mathcal{S} \rangle$, T a tree spanning V and $V' \subseteq V$, the subgraph of T induced by V', denoted by $\mathbf{T}[\mathbf{V'}]$, is defined to contain all vertices of V' and all the edges of T whose both endpoints are in V'.

Definition 6.10. Given a hypergraph $H = \langle V, \mathcal{S} \rangle$:

- G_{ES} is the weighted graph with vertex set V_{ES} and edge set E_{ES}, where $V_{ES} \equiv V$ and E_{ES} contains edge (v, u) (for $v \neq u$) if there exists a cluster S_i such that $\{v, u\} \subseteq S_i$.
- For every edge $(v, u) \in E_{ES}$ and every cluster $S_i \in \mathcal{S}$:

$$w_i(v, u) = \begin{cases} 1 & \text{if } v, u \in S_i \\ 0 & \text{otherwise} \end{cases}$$

- For every edge $(v,u) \in E_{ES} : w(v,u) = \sum_{i=1}^{m} w_i(v,u) = |\{S_i : i \in \{1,\ldots,m\}, \{v,u\} \subseteq S_i\}|$.
- For every tree T of G_{ES} and every cluster $S_i \in \mathcal{S} : w_i(T) = \sum_{(v,u)\in E(T)} w_i(v,u)$, where $E(T)$ is the set of edges in T.
- For every tree T of G_{ES}: $w(T) = \sum_{i=1}^{m} w_i(T)$.

The following theorem is proved by Guttmann-Beck, Sorek and Stern [6].

Theorem 6.11. Given a hypergraph $H = \langle V, \mathcal{S}\rangle$ and T_{ES} a maximum spanning tree of G_{ES}, $w(T_{ES}) = \sum_{i=0}^{m} |S_i| - m$ if and only if H has a feasible solution tree, in this case T_{ES} is a feasible solution tree.

A second proof for Lemma 6.8. The idea of the proof is as follows. Let $H = \langle V, \mathcal{S}\rangle$ such that H satisfies the Helly property and $G_{int}(\mathcal{S}) = G$ is a 2×1 one sided clique grid graph. Next, we will present IL, a feasible vertices insertion list of H, such that IL adds the chords addition list of chords $(s_{0,0}, s_{1,1})$ and $(s_{1,1}, s_{2,0})$ to $G_{int}(\mathcal{S})$. According to Theorem 3.2, $G_{int}(\mathcal{S} + IL)$ is chordal, and therefore, $AL = \{(s_{0,0}, s_{1,1}), (s_{1,1}, s_{2,0})\}$ is a feasible chords addition list of G.

Since H satisfies the Helly property, the following nodes exist: $u_1 \in S_{0,0} \cap S_{1,0} \cap S_{2,0}$, $u_2 \in S_{0,0} \cap S_{0,1}$, $u_3 \in S_{2,0} \cap S_{2,1}$. We construct the following vertices insertion list: $IL = \{(u_1, S_{1,1}), (u_2, S_{1,1}), (u_3, S_{1,1})\}$. Denote $S'_{1,1} = S_{1,1} \cup \{u_1, u_2, u_3\}$. We prove that IL is a feasible vertices insertion list, according to Theorem 6.11, by constructing G_{ES} and a maximum spanning tree T_{ES}.

In the following discussion we refer to clusters in $\mathcal{S}$, before inserting $\{u_1, u_2, u_3\}$ to $S_{1,1}$. Denote:

- $R_{(0,0),(1,0),(2,0)} = S_{0,0} \cap S_{1,0} \cap S_{2,0}$.
- For $S_i \neq S_j$ denote $R_{i,j} = S_i \cap S_j \setminus \bigcup_{S\in\mathcal{S}, S\neq S_i, S\neq S_j} S$, the set of vertices contained only in $S_i \cap S_j$. For example, $R_{(0,0)(1,0)} = S_{0,0} \cap S_{1,0} \setminus S_{2,0}$.
- For $S_i \in \mathcal{S}$ denote $R_i = S_i \setminus \bigcup_{S\in\mathcal{S}, S\neq S_i} S$, the set of vertices contained only in S_i.

Note that, $u_1 \in R_{(0,0),(1,0),(2,0)}, u_2 \in R_{(0,0),(0,1)}, u_3 \in R_{(2,0),(2,1)}$.

Create the following subtrees: $T_{(0,0),(1,0),(2,0)}$ spanning $R_{(0,0),(1,0),(2,0)} \setminus u_1$, $T_{(0,0),(0,1)}$ spanning $R_{(0,0),(0,1)} \setminus u_2$, $T_{(2,0),(2,1)}$ spanning $R_{(2,0),(2,1)} \setminus u_3$. For $\{i,j\} \neq \{(0,0)(0,1)\}$ and $\{i,j\} \neq \{(2,0)(2,1)\}$, $T_{i,j}$ spanning $R_{i,j}$ and T_i spanning R_i for every $S_i \in \mathcal{S}$.

Next, create T_{ES}, described in Figure 49, by connecting all these subtrees and vertices u_1, u_2, u_3.

$T_{(0,0),(1,0),(2,0)}$ contains $|R_{(0,0),(1,0),(2,0)}| - 1$ vertices, and therefore, $|R_{(0,0),(1,0),(2,0)}| - 2$ edges. The weight of each edge inside this subtree is 3.

$T_{(0,0),(0,1)}$ contains $|R_{(0,0),(0,1)}| - 1$ vertices, and therefore, $|R_{(0,0),(0,1)}| - 2$ edges. The weight of each edge inside this subtree is 2.

$T_{(2,0),(2,1)}$ contains $|R_{(2,0),(2,1)}| - 1$ vertices, and therefore, $|R_{(2,0),(2,1)}| - 2$ edges. The weight of each edge inside this subtree is 2.

For $\{i,j\} \neq \{(0,0)(0,1)\}$ and $\{i,j\} \neq \{(2,0)(2,1)\}$, $T_{i,j}$ contains $|R_{i,j}|$ vertices, and therefore $|R_{i,j}|-1$ edges. The weight of each edge inside this subtree is 2.

For every $S_i \in \mathcal{S}$, T_i contains $|R_i|$ vertices, and therefore $|R_i|-1$ edges. The weight of each edge inside this subtree is 1.

The weight of the edges in T_{ES} which touch u_1, u_2 and u_3 is described in Figure 49.

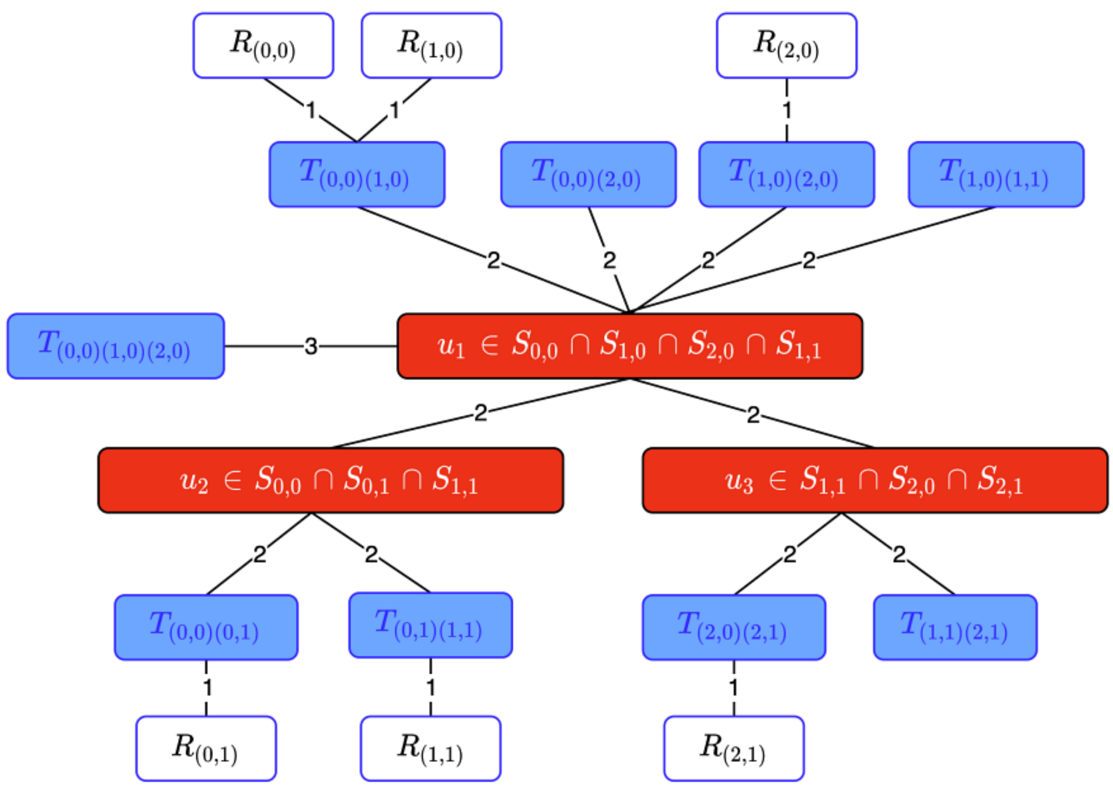


Figure 49: A 2×1 one sided clique grid graph - constructing generic solution tree

The weight of the tree is:

$$
\begin{aligned}
w(T_{ES}) = {} & 3 * (|R_{(0,0),(1,0),(2,0)}| - 2) \\
& + 2 * [(|R_{(0,0),(0,1)}| - 2 + |R_{(2,0),(2,1)}| - 2 + |R_{(0,0),(1,0)}| - 1 + \ldots + |R_{(1,1),(2,1)}| - 1)] \\
& + 1 * (|R_{0,0}| - 1 + |R_{0,1}| - 1 + \ldots + |R_{2,1}| - 1) \\
& + (3 * 1 + 2 * 10 + 1 * 6) \\
& = 3 * |R_{(0,0),(1,0),(2,0)}| + 2 * (|R_{(0,0),(0,1)}| + \ldots + |R_{(2,0),(2,1)}|) + 1 * (|R_{0,0}| \ldots + |R_{2,1}|) \\
& - (3 * 2 + 2 * (2 + 2 + 6) + 1 * 6) + (3 * 1 + 2 * 10 + 1 * 6) \\
& = |S_{0,0}| + |S_{1,0}| + |S_{2,0}| + |S_{0,1}| + |S_{1,1}| + |S_{2,1}| - 3 \\
& = |S_{0,0}| + |S_{1,0}| + |S_{2,0}| + |S_{0,1}| + |S'_{1,1}| + |S_{2,1}| - 6 \\
& = \sum_{S_i \in \mathcal{S}+IL}^{m} |S_i| - 6 = \sum_{S_i \in \mathcal{S}+IL}^{m} |S_i| - m
\end{aligned}
$$

According to Theorem 6.11, since the weight of T_{ES} is $\sum_{i=1}^{m} |S_i| - m$, T_{ES} is a maximum spanning tree for G_{ES} and is also a feasible solution tree for $H+IL$. According to Theorem 3.2, $G_{int}(\mathcal{S}+IL)$ is chordal. $G_{int}(\mathcal{S}+IL)$ is the graph presented in Figure 48, which is $G+AL$. Therefore, AL is a feasible chords addition list. $\square$

Lemma 6.12. Let $G = \langle U, E \rangle$ be a 2×1 one sided clique grid graph. $mAL(G) = 2$.

Proof. G contains the following simple cycle: $C_1 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{2,0} - s_{0,0}$. C_1 is a five node cycle. According to Theorem 6.1, at least two chords are required to achieve chordality on C_1. Let $AL = \{(s_{0,0}, s_{1,1}), (s_{1,1}, s_{2,0})\}$. According to Lemma 6.8, AL is a feasible chords addition list, such that $|AL| = 2$. Therefore, $mAL(G) = 2$. $\square$

6.2.2 3×1 One sided clique grid graphs

In this section we present minimum cardinality feasible chords addition list for 3×1 one sided clique grid graphs.

Lemma 6.13. Let $G = \langle U, E \rangle$ be a 3×1 one sided clique grid graph.
Let $AL = \{(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1}), (s_{1,0}, s_{2,1}), (s_{3,0}, s_{2,1})\}$. AL is a feasible chords addition list of G.

Proof. G contains the following cycles, see Figure 50:

- Three basic cycles $C_1 = s_{0,0} - s_{0,1} - s_{1,1} - s_{1,0} - s_{0,0}$, $C_2 = s_{1,0} - s_{1,1} - s_{2,1} - s_{2,0} - s_{1,0}$, $C_3 = s_{2,0} - s_{2,1} - s_{3,1} - s_{3,0} - s_{2,0}$. Chords $(s_{0,0}, s_{1,1})$, $(s_{1,0}, s_{2,1})$ and $(s_{3,0}, s_{2,1})$ are chords of C_1, C_2 and C_3, respectively, achieving chordality in each cycle.

- A cycle $C_4 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{0,0}$. Chords $(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1}), (s_{2,1}, s_{3,0})$ achieve chordality in C_4.

- A cycle $C_5 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1}, s_{2,0} - s_{0,0}$. Chords $(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1})$ achieve chordality in C_5.

- A cycle $C_6 = s_{1,0} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{1,0}$. Chords $(s_{1,0}, s_{2,1}), (s_{3,0}, s_{2,1})$ achieve chordality in C_6.

$C_7 = s_{0,0} - s_{1,1} - s_{2,1} - s_{1,0} - s_{0,0}$ is a new cycle created in $G + AL$. However, edge $(s_{1,0}, s_{1,1})$ is a chord of this cycle. Any of the newly created cycles in $G + AL$ are sub cycles of one of the cycles in G and therefore, is also chordal. $\square$

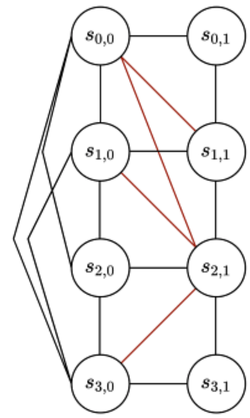


Figure 50: A 3×1 one sided clique grid graph. The edges of G are colored black and the edges in AL are colored red.

Lemma 6.14. Let $G = \langle U, E \rangle$ be a 3×1 one sided clique grid graph. $mAL(G) = 4$.

Proof. G contains a cycle $C_1 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{0,0}$ which is a cycle containing six nodes. According to Theorem 6.1, at least 3 chords are required to achieve chordality on C_1. In addition, G contains the cycle $C_2 = s_{1,0} - s_{1,1} - s_{2,1} - s_{2,0} - s_{1,0}$ which is a cycle containing four nodes. According to Theorem 6.1, at least one chord is required to achieve chordality on C_2. The chord of C_2 cannot be a chord of C_1. Therefore, $mAL(G) \geq 3 + 1 = 4$. According to Lemma 6.13, there is a feasible chords addition list with cardinality 4. Hence, $mAL(G) = 4$. $\square$

6.2.3 4×1 One sided clique grid graphs

In this section we present minimum cardinality feasible chords addition list for 4×1 one sided clique grid graphs.

First, we specify all the cycles in this graph. Figure 51 presents a 4×1 one sided clique grid graph.

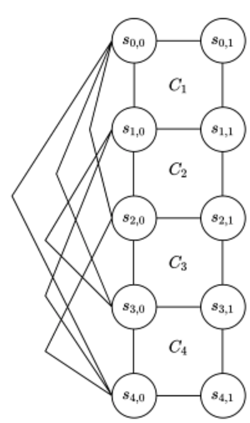
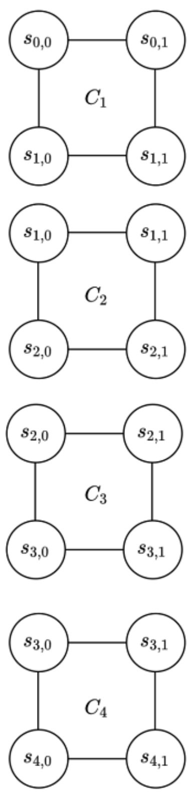


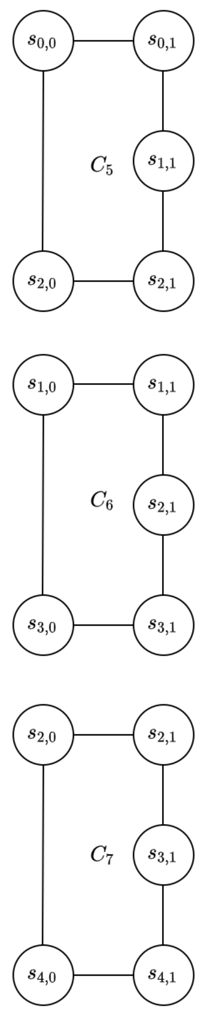
Figure 51: A 4×1 one sided clique grid graph

Let $G = \langle U, E \rangle$ be a 4×1 one sided clique grid graph. G contains the following cycles:

- Four basic cycles $C_1 = s_{0,0} - s_{0,1} - s_{1,1} - s_{1,0} - s_{0,0}$, $C_2 = s_{1,0} - s_{1,1} - s_{2,1} - s_{2,0} - s_{1,0}$, $C_3 = s_{2,0} - s_{2,1} - s_{3,1} - s_{3,0} - s_{2,0}$, $C_4 = s_{3,0} - s_{3,1} - s_{4,1} - s_{4,0} - s_{3,0}$, see Figure 52a.
- Three 5 node cycles $C_5 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{2,0} - s_{0,0}$, $C_6 = s_{1,0} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{1,0}$, $C_7 = s_{2,0} - s_{2,1} - s_{3,1} - s_{4,1} - s_{4,0} - s_{2,0}$, see Figure 52b.
- Two 6 node cycles $C_8 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{0,0}$, $C_9 = s_{1,0} - s_{1,1} - s_{2,1} - s_{3,1} - s_{4,1} - s_{4,0} - s_{1,0}$, see Figure 53a.
- One 7 node cycle $C_{10} = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{3,1} - s_{4,1} - s_{4,0} - s_{0,0}$, see Figure 53b.

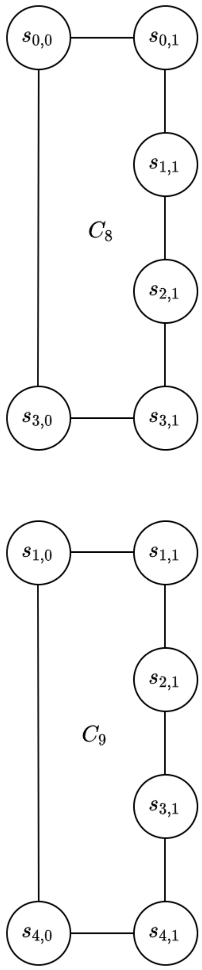


(a) Cycles C_1, C_2, C_3, C_4 of size 4

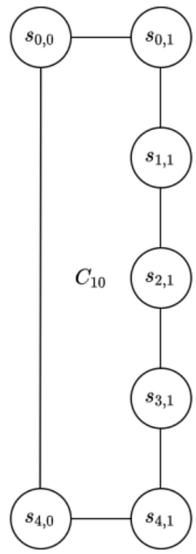


(b) Cycles C_5, C_6, C_7 of size 5

Figure 52: Cycles of size 4 and 5



(a) Cycles C_8, C_9 of size 4



(b) Cycle C_{10} of size 7

Figure 53: Cycles of size 6 and 7

Lemma 6.15. Let $G = \langle U, E \rangle$ be a 4×1 one sided clique grid graph.
Let $AL = \{(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1}), (s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0}), (s_{2,1}, s_{4,0}), (s_{3,1}, s_{4,0})\}$. AL is a feasible chords addition list of G.

Proof. G contains the following cycles $C, \ldots, C_{10}$ described in the beginning of 6.2.3, see Figure 54.

- Chords $(s_{0,0}, s_{1,1})$, $(s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0})$ and $(s_{3,1}, s_{4,0})$ are chords of C_1, C_2, C_3 and C_4, respectively, achieving chordality in each cycle.
- Chords $(s_{0,0}, s_{1,1})$ and $(s_{0,0}, s_{2,1})$ are chords of C_5 achieving chordality in each cycle. Chords $(s_{1,0}, s_{2,1})$ and $(s_{2,1}, s_{3,0})$ are chords of C_6 achieving chordality in each cycle. Chords $(s_{2,1}, s_{4,0})$ and $(s_{3,1}, s_{4,0})$ are chords of C_7 achieving chordality in each cycle.
- Chords $(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1})$ and $(s_{2,1}, s_{3,0})$ are chords of C_8 achieving chordality in each cycle. Chords $(s_{1,0}, s_{2,1}), (s_{2,1}, s_{4,0})$ and $(s_{3,1}, s_{4,0})$ are chords of C_9 achieving chordality in each cycle.

- Chords $(s_{0,0}, s_{1,1})$, $(s_{0,0}, s_{2,1}), (s_{2,1}, s_{4,0})$ and $(s_{3,1}, s_{4,0})$ are chords of C_{10} achieving chordality in C_{10}.

$C_{11} = s_{0,0} - s_{1,1} - s_{2,1} - s_{1,0} - s_{0,0}, C_{12} = s_{2,1} - s_{3,1} - s_{4,0} - s_{3,0} - s_{2,1}$ are two new cycles created in $G + AL$. However, edges $(s_{1,0}, s_{1,1}), (s_{3,0}, s_{3,1})$ are chords of these cycles respectively. Any other newly created cycle in $G + AL$ is a sub cycle of one of the cycles in G and therefore, is also chordal. □

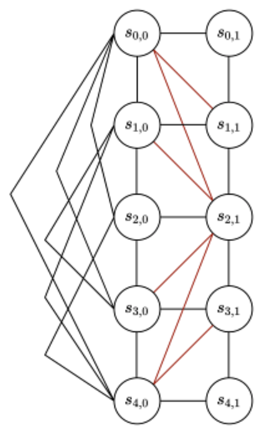


Figure 54: A 4×1 one sided clique grid graph. The edges of G are colored black and the edges in AL are colored red.

Lemma 6.16. Let $G = \langle U, E \rangle$ be a 4×1 one sided clique grid graph. $mAL(G) = 6$.

Proof. G contains cycle C_{10} which is a cycle containing seven nodes. According to Theorem 6.1, at least four chords are required to achieve chordality on C_{10}. In addition, G contains two cycles C_2 and C_3 which are cycles containing four nodes. According to Theorem 6.1, at least one chord is required to achieve chordality on C_2 and at least one chord is required to achieve chordality on C_3. The chords of C_2, C_3 cannot be chords of C_{10}. Therefore, $mAL(G) \geq 4 + 1 + 1 = 6$. According to Lemma 6.15, there is a feasible chords addition list with cardinality 6. Hence, $mAL(G) = 6$. □

Let $G = \langle U, E \rangle$ be a 4×1 one sided clique grid graph, see Figure 51. Let $AL = \{(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1}), (s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0}), (s_{2,1}, s_{4,0}), (s_{3,1}, s_{4,0})\}$. In Lemma 6.15, we prove that AL is a feasible chords addition list and Lemma 6.16 proves it is a minimum cardinality feasible chords addition list. We check whether there are other minimum cardinality feasible chords addition lists, hence AL is not unique.

The procedure we use to check whether AL is unique or not is using a software that checks all possible chords addition lists of size six for G. Appendix A presents Python code implementing this test. The code uses a library called Networkx [8]. This library provides a function which checks whether a graph G is chordal. The algorithm used to test chordality is described [13].

We first notice that any basic cycle needs at least one chord, and have two options for this chord. Hence, there are $2^4 = 16$ options for the chords of the basic cycles.

If the chords addition list contains 6 chords, it may contain 2 chords outside the basic cycles. There are $\frac{\binom{10}{2}\binom{8}{2}}{2} = 630$ options to choose these 2 edges. Altogether, there are $16 * 630 = 10,080$ options to consider.

Our algorithm:

- For every possible chords addition list:
  - Generate chords addition list AL_i.
  - Check whether $G + AL_i$ is chordal, using the procedure provided [13].

The result of our test is that $AL = \{(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1}), (s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0}), (s_{2,1}, s_{4,0}), (s_{3,1}, s_{4,0})\}$ is the only feasible chords addition list.

We want to further understand why there is only one option for a minimum cardinality feasible chords addition list. First we study the chords added to the basic cycles.

Let $G = \langle U, E\rangle$ be a 4×1 one sided clique grid graph. G contains the following cycles $C_, \ldots, C_{10}$, described in the beginning of Section 6.2.3. Consider the simple cycle C_1. There are two options for the chord which achieves chordality in this cycle. Either it is $(s_{0,0}, s_{1,1})$, which is the chord used in Lemma 6.15, or it is $(s_{0,1}, s_{1,0})$.

Lemma 6.17. Let $G = \langle U, E\rangle$ be a 4×1 one sided clique grid graph. If a feasible chords addition list AL contains chord $(s_{0,1}, s_{1,0})$ then $|AL| \geq 7$. Hence, AL is not a minimum cardinality chords addition list.

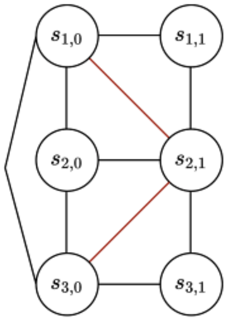
Proof. G contains cycle C_{10}, which is a seven node cycle. According to Theorem 6.1, AL contains at least four chords to achieve chordality in C_{10}, neither of these chords is $(s_{0,1}, s_{1,0})$. In addition, AL must contain at least one chord to achieve chordality in C_2, and one in C_3. Obviously, these two chords are two different chords, which are not $(s_{0,1}, s_{1,0})$ and not the chords used in C_{10}. Hence, $|AL| \geq 1 + 4 + 1 + 1 \geq 7$. □

Corollary 6.18. Let $G = \langle U, E\rangle$ be a 4×1 one sided clique grid graph. If a feasible chords addition list AL contains $(s_{3,0}, s_{4,1})$ then $|AL| \geq 7$ and AL is not a minimum cardinality feasible chords addition list.

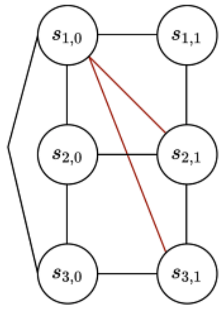
According to Lemma 6.16, Lemma 6.17 and Corollary 6.18, we conclude that if AL is a minimum cardinality feasible chords addition list, it must contain $(s_{0,0}, s_{1,1})$ and $(s_{3,1}, s_{4,0})$. Next, we consider the chords to add to C_2 and C_3.

Lemma 6.19. Let $G = \langle U, E\rangle$ be a 4×1 one sided clique grid graph. If AL is a feasible chords addition list of G and $|AL| = 6$, then $\{(s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0})\} \subset AL$.

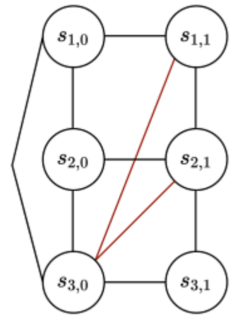
Proof. Consider $C_6 = s_{1,0} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{1,0}$. This cycle contains 5 nodes and according to Theorem 6.1, at least 2 edges should be added to this cycle to achieve chordality. The following drawings present all the possibilities of choosing 2 chords for C_6.



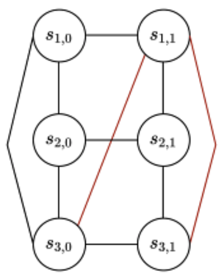
(a) Option 1



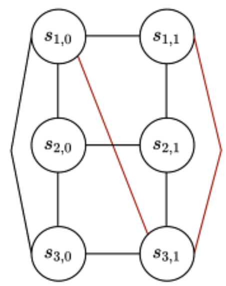
(b) Option 2



(c) Option 3



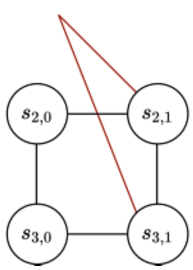
(d) Option 4



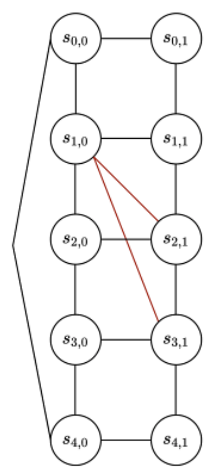
(e) Option 5

Figure 55: Adding two chords to C_6 - all options

Suppose Option b in Figure 55 is correct (a similar argument holds also for Option c). In this case, $\{(s_{1,0}, s_{2,1}), (s_{1,0}, s_{3,1})\} \in AL$. Note that neither $(s_{1,0}, s_{2,1})$ nor $(s_{1,0}, s_{3,1})$ can be used as chords of C_3 or C_{10}, see Figure 56.



(a) Cycle C_3

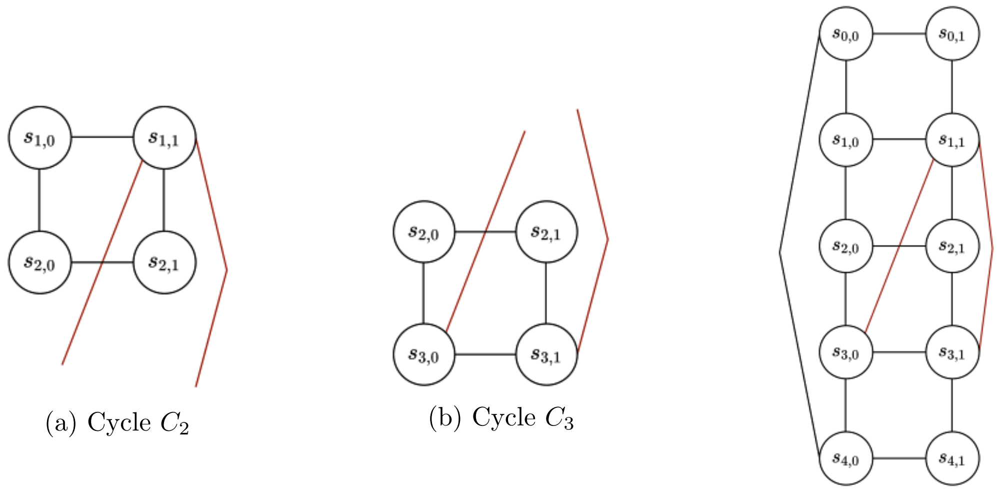


(b) Cycle C_{10}

Figure 56: Adding chords $(s_{1,0}, s_{2,1})$ and $(s_{1,0}, s_{3,1})$

According to Theorem 6.1, AL contains at least 1 more chord to achieve chordality in C_3 and at least 4 more chords to achieve chordality in C_{10}. Hence, $|AL| \geq 7$, contradicting the assumption of the Lemma.

Suppose Option d is correct (a similar argument holds also for option e). In this case, $\{(s_{1,1}, s_{3,0}), (s_{1,1}, s_{3,1})\} \in AL$. Note that neither $(s_{1,1}, s_{3,0})$ nor $(s_{1,1}, s_{3,1})$ can be used as chords of C_2 or C_3. However, $(s_{1,1}, s_{3,1})$ is a chord of C_{10}, see Figure 57.



(a) Cycle C_2

(b) Cycle C_3

(c) Cycle C_{10}

Figure 57: Adding chords $(s_{1,1}, s_{3,0})$ and $(s_{1,1}, s_{3,1})$

According to Theorem 6.1, AL contains at least 3 more chords to achieve chordality in C_{10}, 1 chord to achieve chordality in C_2 and 1 chord to achieve chordality in C_3. By the structure of C_2, C_3 and C_{10}, these chords are different and therefore $|AL| \geq 7$, contradicting the assumption of the Lemma.

Hence, Option a is correct and $\{(s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0})\} \subset AL$. □

Theorem 6.20. Let $G = \langle U, E\rangle$ be a 4×1 one sided clique grid graph. Let $AL = \{(s_{0,0}, s_{1,1}), (s_{0,0}, s_{2,1}), (s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0}), (s_{2,1}, s_{4,0}), (s_{3,1}, s_{4,0})\}$. AL is the only minimum cardinality feasible chords addition list.

Proof. Consider AL a minimum cardinality feasible chords addition list.

According to Lemma 6.16, $|AL| = 6$. According to Lemma 6.19 and Corollary 6.18, $AL' = \{(s_{0,0}, s_{1,1}), (s_{1,0}, s_{2,1}), (s_{2,1}, s_{3,0}), (s_{3,1}, s_{4,0})\} \subset AL$, see Figure 58.

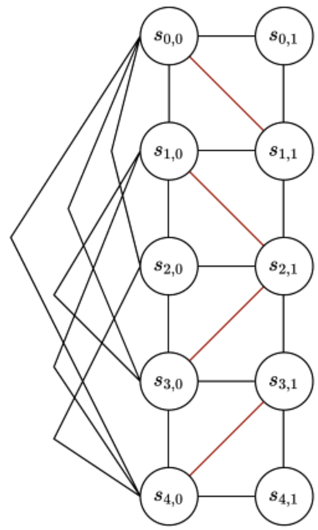


Figure 58: $G + AL'$ for proof of Theorem 6.20. The edges of G are colored black and the chords in AL' are colored red.

$G + AL$ contains the following cycles: $C' = s_{0,0} - s_{1,1} - s_{2,1} - s_{2,0} - s_{0,0}$ which contains 4 nodes, $C'' = s_{2,0} - s_{2,1} - s_{3,1} - s_{4,0} - s_{2,0}$ which contains 4 nodes, $C''' = s_{0,0} - s_{1,1} - s_{2,1} - s_{3,1} - s_{4,0} - s_{0,0}$ which contains 5 nodes.

According to Theorem 6.1, to achieve chordality in C', AL contains at least one chord which may be either $(s_{0,0}, s_{2,1})$ or $(s_{1,1}, s_{2,0})$. If AL contains $(s_{1,1}, s_{2,0})$, which is not a chord of C''', according to Theorem 6.1, AL adds at least 2 chords to C'''. In this case, AL contains the 4 chords of AL', $(s_{1,1}, s_{2,0})$ and 2 more chords of C''', contradicting the assumption of the Lemma that $|AL| = 6$.

A similar argument also proves that AL contains $(s_{2,1}, s_{4,0})$ as a chord of C''.

Hence, $AL = AL' \cup \{(s_{0,0}, s_{2,1}), (s_{2,1}, s_{4,0})\}$. Lemma 6.15 ensures that this is indeed a feasible chords addition list of G. Thus, AL is a unique minimum cardinality feasible chords addition list. $\square$

6.2.4 Chords addition lists of even $n \times 1$ one sided clique grid graphs

In this section we present feasible chords addition lists for $n \times 1$ one sided clique grid graphs.

Let $G = \langle U, E \rangle$ be an $n \times 1$ one sided clique grid graph for an even n, $n \geq 2$, as described in Figure 59.

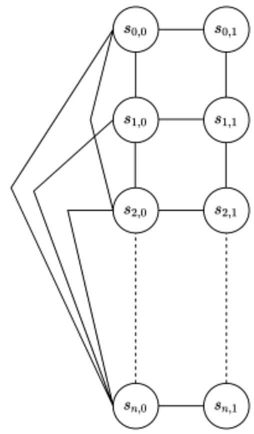


Figure 59: An $n \times 1$ one sided clique grid graph

Definition 6.21. Let $AL_{up} = \{(s_{i,0}, s_{j,1}) \mid 0 \leq i \leq \frac{n}{2} - i, i+1 \leq j \leq \frac{n}{2}\}$, $AL_{down} = \{(s_{i,0}, s_{j,1}) \mid \frac{n}{2} + 1 \leq i \leq n, \frac{n}{2} \leq j \leq i-1\}$, $AL = AL_{up} \cup AL_{down}$.

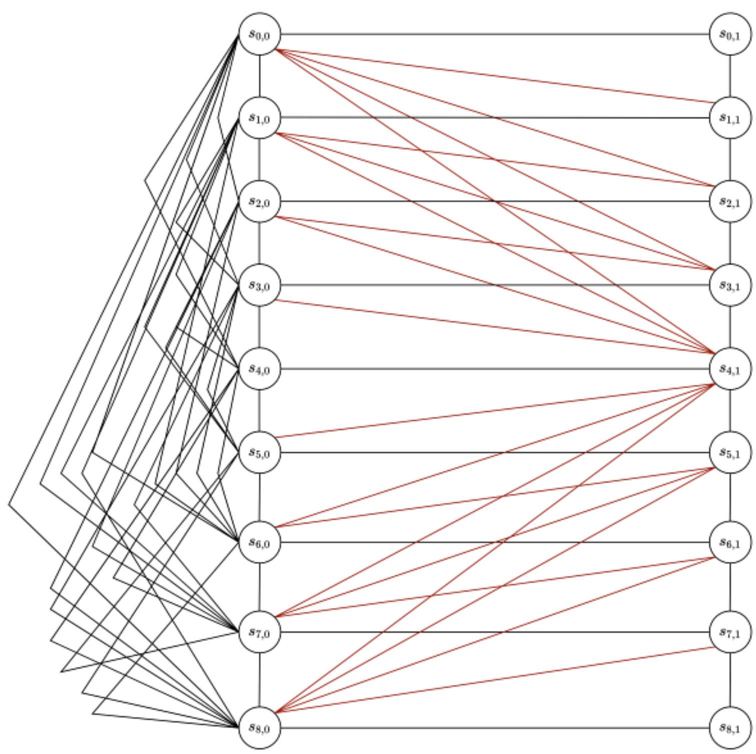


Figure 60: An 8×1 one sided clique grid graph, where the edges of G are colored black and the chords in AL are colored red

Since $|AL_{up}| = |AL_{down}|$,

$$
\begin{aligned}
|AL| &= 2 * \sum_{i=0}^{\frac{n}{2}-1} (\frac{n}{2} - i) \\
&= 2 * \Big(\sum_{i=0}^{\frac{n}{2}-1} (\frac{n}{2}) - \sum_{i=0}^{\frac{n}{2}-1} (i) \Big) \\
&= 2 * \Big(\frac{n}{2} * \frac{n}{2} - \frac{\frac{n}{2} * (\frac{n}{2} - 1)}{2} \Big) \\
&= 2 * \Big(\frac{n^2}{4} - (\frac{n^2}{8} - \frac{n}{4}) \Big) \\
&= 2 * (\frac{n^2}{8} + \frac{n}{4}) \\
&= \frac{n^2}{4} + \frac{n}{2}
\end{aligned}
$$

Note that $|AL|$ agrees with all the sizes of minimum cardinality chords addition lists presented in sections 6.2.1 and 6.2.3.

If $n = 2$, $|AL| = \frac{n^2}{4} + \frac{n}{2} = \frac{2^2}{4} + \frac{2}{2} = 1 + 1 = 2$. According to Lemma 6.12, in this case $mAL(G) = 2$.

If $n = 4$, $|AL| = \frac{n^2}{4} + \frac{n}{2} = \frac{4^2}{4} + \frac{4}{2} = 4 + 2 = 6$. According to Lemma 6.16, in this case $mAL(G) = 6$.

If $n = 6$, $|AL| = \frac{n^2}{4} + \frac{n}{2} = \frac{6^2}{4} + \frac{6}{2} = 9 + 3 = 12$.

If $n = 8$, $|AL| = \frac{n^2}{4} + \frac{n}{2} = \frac{8^2}{4} + \frac{8}{2} = 16 + 4 = 20$.

Lemma 6.22. Let $G = \langle U, E \rangle$ be an $n \times 1$ one sided clique grid graph for an even n, $n \geq 2$. AL, defined in Definition 6.21, is a feasible chords addition list.

Proof. Assume by contradiction that a chordless cycle C exists in $G+AL$. We consider the following possible places for C. Let $U_l = \{s_{0,0}, \ldots, s_{n,0}\}$, $U_r = \{s_{0,1}, \ldots, s_{n,1}\}$.

- C is a basic cycle, $s_{i,0} - s_{i,1} - s_{i+1,1} - s_{i+1,0} - s_{i,0}$, for $0 \leq i \leq n-1$. If $i \leq \frac{n}{2} - 1$, then AL contains $(s_{i,0}, s_{i+1,1})$ which is a chord. If $i \geq \frac{n}{2}$, then AL contains $(s_{i+1,0}, s_{i,1})$ which is a chord.

- C contains only nodes from U_l, $C = s_{i_1,0} - \ldots - s_{i_k,0} - s_{i_1,0}$. However, according to the structure of G, it contains a clique induced on $\{s_{i_1,0}, \ldots, s_{i_k,0}\}$, contradicting the assumption that C is chordless.

- C contains only nodes from U_r, $C = s_{j_1,1} - \ldots - s_{j_k,1} - s_{j_1,1}$. However, by the structure of $G + AL$, there are no chords $(s_{j_1,1}, s_{j_k,1})$ with $j_k > j_1 + 1$. Thus, there is no cycle of this structure.

- C contains at least one node from U_r and at least one node from U_l.

  Assume by contradiction, that C contains three nodes from U_l. G induces a clique on these nodes, contradicting the assumption that C is chordless. Therefore, C contains one node or two nodes from U_l. Since C contains at least four nodes, it contains at least two nodes from U_r, denote these nodes as $s_{j_1,1}$ and $s_{j_2,1}$, with $j_1 < j_2$. According to the structure of $G+AL$, C includes all nodes $s_{j_1,1}, \ldots, s_{j_2,1}$. Since G contains a clique on the nodes in U_l and C contains two nodes from U_l, they are adjacent in the cycle. Therefore, we can assume that C is composed from a one or two connected nodes from U_l and a path $s_{j_1}, \ldots, s_{j_2}$, such that every node in this path is from U_r .Without loss of generality, suppose that $j_1 < j_2$ and that s_{i_1} is the node from U_l that is adjacent to s_{j_1}.

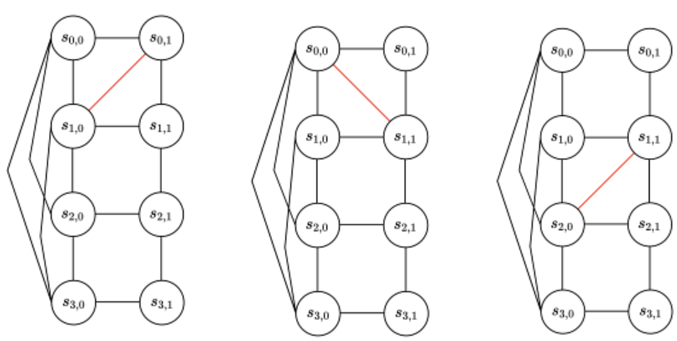
We consider three options:

- $j_1 < j_2 < \frac{n}{2}$. Consider the node in C which is from U_l and appears in C before $s_{j_1,1}$, thus C contains $(s_{i_1,0}, s_{j_1,1})$. By Definition 6.21 of AL and since $j_1 < \frac{n}{2}$, it follows that $i_1 < j_1$. Therefore, C contains $(s_{i_1,0}, s_{i_1+1,1})$ which is a chord of C.
- $j_2 > j_1 > \frac{n}{2}$. Consider the node in C which is from U_l and appears in C before $s_{j_1,1}$, thus C contains $(s_{i_1,0}, s_{j_1,1})$. By Definition 6.21 of AL and since $j_1 > \frac{n}{2}$, it follows that $i_1 > j_1$. Therefore, C contains $(s_{i_1,0}, s_{i_1-1,1})$ which is a chord of C.
- $j_1 < \frac{n}{2}, j_2 > \frac{n}{2}$. In this case C contains node $s_{\frac{n}{2},1}$. Consider the node in C which is from U_l and appears in C before $s_{j_1,1}$, thus C contains $(s_{i_1,0}, s_{j_1,1})$. By Definition 6.21 of AL and since $j_1 < \frac{n}{2}$, it follows that $i_1 < j_1$. $G + AL$ contains chord $(s_{i_1,0}, s_{\frac{n}{2},1})$ which is a chord of C.

Therefore, a cycle C cannot exist in $G + AL$ and AL is a feasible chords addition list. □

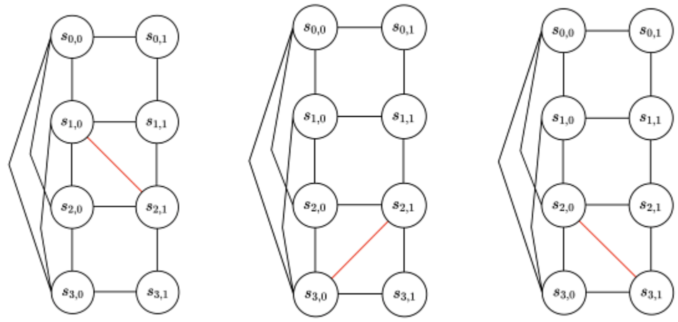
6.2.5 Using linear programming to express the minimum cardinality of chords addition lists

In this section we use linear programming in binary variables to express the problem of finding minimum cardinality feasible chords addition lists. We first demonstrate the problem on a $G = \langle U, E\rangle$, a 3×1 one side clique grid graph. Figures 61 - 65 describe all possible chords that can be added to G. The edges in G are colored black, and the chord is colored red.



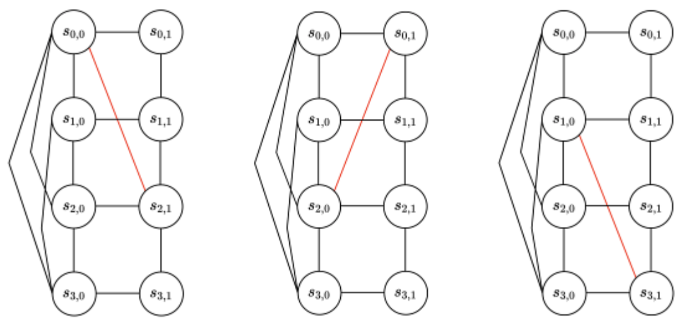
(a) e_1 (b) e_2 (c) e_3

Figure 61: Possible chords in G, edge e_1, e_2 or e_3



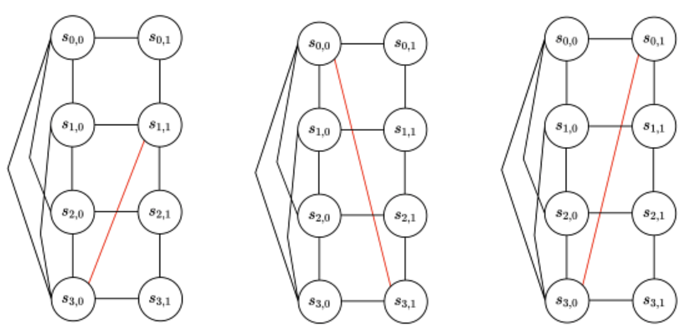
(a) e_4 (b) e_5 (c) e_6

Figure 62: Possible chords in G, edge e_4, e_5 or e_6



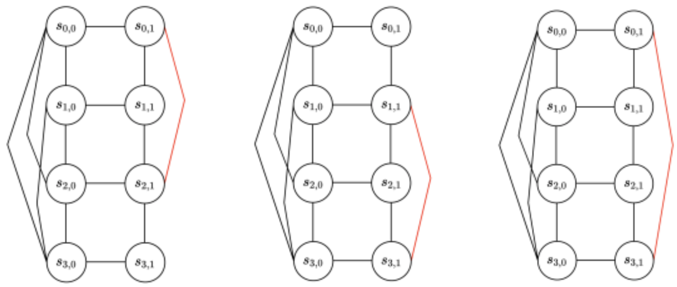
(a) e_7 (b) e_8 (c) e_9

Figure 63: Possible chords in G, edge e_7, e_8 or e_9



(a) e_{10} (b) e_{11} (c) e_{12}

Figure 64: Possible chords in G, edge e_{10}, e_{11} or e_{12}



(a) e_{13} (b) e_{14} (c) e_{15}

Figure 65: Possible chords in G, edge e_{13}, e_{14} or e_{15}

Consider the following edges:

$$e_1 = \{(s_{1,0}, s_{0,1})\}, e_2 = \{(s_{0,0}, s_{1,1})\}$$
$$e_3 = \{(s_{2,0}, s_{1,1})\}, e_4 = \{(s_{1,0}, s_{2,1})\}$$
$$e_5 = \{(s_{3,0}, s_{2,1})\}, e_7 = \{(s_{0,0}, s_{2,1})\}$$
$$e_8 = \{(s_{2,0}, s_{0,1})\}, e_9 = \{(s_{1,0}, s_{3,1})\}$$
$$e_{10} = \{(s_{3,0}, s_{1,1})\}, e_{11} = \{(s_{0,0}, s_{3,1})\}$$
$$e_{12} = \{(s_{3,0}, s_{0,1})\}, e_{13} = \{(s_{0,1}, s_{2,1})\}$$
$$e_{14} = \{(s_{1,1}, s_{3,1})\}, e_{15} = \{(s_{0,1}, s_{3,1})\}$$

G contains the following cycles and chords:

- G contains cycle $C_1 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{0,0}$. C_1 is a six nodes cycle. According to Theorem 6.1, it requires at least three chords to achieve chordality. We define the sets of chords whose size is three, such that their addition to G achieves chordality for C_1. The sets are $C_{1,1} = \{e_2, e_7, e_{11}\}, C_{1,2} = \{e_2, e_{14}, e_{11}\}$, $C_{1,3} = \{e_5, e_{10}, e_{12}\}, C_{1,4} = \{e_5, e_{13}, e_{12}\}$, $C_{1,5} = \{e_2, e_5, e_{10}\}, C_{1,6} = \{e_2, e_{10}, e_{14}\}$.
- G contains two five node cycles, $C_2 = s_{0,0} - s_{0,1} - s_{1,1} - s_{2,1} - s_{2,0} - s_{0,0}$ and $C_3 = s_{1,0} - s_{1,1} - s_{2,1} - s_{3,1} - s_{3,0} - s_{1,0}$. According to Theorem 6.1, each cycle requires the addition of at least two chords to achieve chordality.

  We define the sets of chords whose size is two, such that their addition to G achieves chordality for C_2. The sets are $C_{2,1} = \{e_2, e_3\}, C_{2,2} = \{e_2, e_7\}, C_{2,3} = \{e_3, e_8\}$.

  We define the sets of chords whose size is two, such that their addition to G achieves chordality for C_3. The sets are $C_{3,1} = \{e_4, e_5\}, C_{3,2} = \{e_4, e_9\}, C_{3,3} = \{e_5, e_{10}\}$.
- G contains three basic cycles, $C_4 = S_{0,0} - S_{1,0} - S_{0,1} - S_{1,1} - S_{0,0}$, $C_5 = S_{1,0} - S_{2,0} - S_{1,1} - S_{2,1} - S_{1,0}$, $C_6 = S_{2,0} - S_{3,0} - S_{2,1} - S_{3,1} - S_{2,0}$. According to Theorem 6.1, each cycle requires the addition of at least one chord to achieve chordality.

  We define the sets of chords whose size is one, such that their addition to G achieves chordality for C_4. The sets are $C_{4,1} = \{e_1\}, C_{4,2} = \{e_2\}$.

  We define the sets of chords whose size is one, such that their addition to G achieves chordality for C_5. The sets are C_5 are: $C_{5,1} = \{e_3\}, C_{5,2} = \{e_4\}$.

We define the sets of chords whose size is one, such that their addition to G achieves chordality for C_6. The sets are $C_{6,1} = \{e_5\}, C_{6,2} = \{e_6\}$.

We define the following variables:

$$x_i = \begin{cases} 1 & e_i \text{ is added to G} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{i,j} = \begin{cases} 1 & \text{All the chords of } C_{i,j} \text{ are added to } G \\ & \text{and achieve chordality of cycle } C_i \\ 0 & \text{At least one chord of } C_{i,j} \text{ is not added to } G \end{cases}$$

The linear programming is:

$$
\begin{aligned}
\text{minimize} \sum_{i=1}^{m} x_i \\
\text{s.t. } & x_2 + x_7 + x_{11} \geq 3y_{1,1} = |C_{1,1}|y_{1,1} \\
& x_2 + x_{14} + x_{11} \geq 3y_{1,2} = |C_{1,2}|y_{1,2} \\
& x_5 + x_{10} + x_{12} \geq 3y_{1,3} = |C_{1,3}|y_{1,3} \\
& x_5 + x_{13} + x_{12} \geq 3y_{1,4} = |C_{1,4}|y_{1,4} \\
& x_2 + x_5 + x_{10} \geq 3y_{1,5} = |C_{1,5}|y_{1,5} \\
& x_2 + x_{10} + x_{14} \geq 3y_{1,6} = |C_{1,6}|y_{1,6} \\
& x_2 + x_3 \geq 2y_{2,1} = |C_{2,1}|y_{2,1} \\
& x_2 + x_7 \geq 2y_{2,2} = |C_{2,2}|y_{2,2} \\
& x_3 + x_8 \geq 2y_{2,3} = |C_{2,3}|y_{2,3} \\
& x_4 + x_5 \geq 2y_{3,1} = |C_{3,1}|y_{3,1} \\
& x_4 + x_9 \geq 2y_{3,2} = |C_{3,3}|y_{3,2} \\
& x_5 + x_{10} \geq 2y_{3,3} = |C_{3,3}|y_{3,3} \\
& x_1 + \geq y_{4,1} = |C_{4,1}|y_{4,1} \\
& x_2 \geq y_{4,2} = |C_{4,2}|y_{4,2} \\
& x_3 + \geq y_{5,1} = |C_{5,1}|y_{5,1} \\
& x_4 \geq y_{5,2} = |C_{5,2}|y_{5,2} \\
& x_5 + \geq y_{6,1} = |C_{6,1}|y_{6,1} \\
& x_6 \geq y_{6,2} = |C_{6,2}|y_{6,2} \\
& y_{1,1} + y_{1,2} + y_{1,3} + y_{1,4} + y_{1,5} + y_{1,6} \geq 1 \\
& y_{2,1} + y_{2,2} + y_{2,3} \geq 1 \\
& y_{3,1} + y_{3,2} + y_{3,3} \geq 1 \\
& y_{4,1} + y_{4,2} \geq 1 \\
& y_{5,1} + y_{5,2} \geq 1 \\
& y_{6,1} + y_{6,2} \geq 1 \\
& x_i \in \{0,1\}, 1 \leq i \leq 15 \\
& \forall i, \forall j, y_{i,j} \in \{0,1\}
\end{aligned}
$$

Note that the constraint $x_2+x_7+x_{11} \geq 3y_{1,1} = |C_{1,1}|y_{1,1}$ ensures that if $y_{1,1} = 1$ then all the edges e_2, e_7 and e_{11} are added to G.

Note that the constraint $y_{1,1} + y_{1,2} + y_{1,3} + y_{1,4} + y_{1,5} + y_{1,6} \geq 1$ means that at least one of the sets $\{e_2, e_7, e_{11}\}, \{e_2, e_{14}, e_{11}\}$, $\{e_5, e_{10}, e_{12}\}, \{e_5, e_{13}, e_{12}\}$, $\{e_2, e_5, e_{10}\}$ or $\{e_2, e_{10}, e_{14}\}$, which causes chordality of C_1, is added to G.

6.2.6 Linear programming for $n \times 1$ one sided clique grid graphs

In this section we generalize the calculation for $n \times 1$ one sided clique grid graphs.

Let $G = \langle U, E \rangle$ be an $n \times 1$ one sided clique grid graph for even n, $n \geq 2$. G contains k cycles, $C_1, \ldots, C_k$.

There is one $n{+}3$ nodes cycle, two $n{+}2$ node cycle, $\ldots$, n basic cycles. Therefore, $k = 1{+}2{+}\ldots{+}n = \frac{n(n+1)}{2}$. According to Theorem 6.1, a cycle whose size is l requires at least $l-3$ chords to achieve chordality. For a cycle C_i, we define set $C_{i,l_1}, \ldots, C_{i,l_n}$.

We get a set of sets, each of which includes $|C_i| - 3$ chords. Adding the chords of $C_{i,j}$ achieves chordality of cycle C_i. Let l_i be the number of sets whose addition to cycle C_i may achieve chordality in C_i.

We define the following variables:

$$x_i = \begin{cases} 1 & e_i \text{ is added to G} \\ 0 & \text{otherwise} \end{cases}$$

$$y_{i,j} = \begin{cases} 1 & \text{All the chords of } C_{i,j} \text{ are added to } G \\ & \text{and achieves chordality of cycle } C_i \\ 0 & \text{At least one chord of } C_{i,j} \text{ is not added to } G \end{cases} \quad \text{for } 1 \leq i \leq k, 1 \leq j \leq l_i$$

E' is the list of all possible chords.

The linear programming is:

$$\begin{aligned} \text{minimize} & \sum_{e \in E'} x_e \\ \text{s.t.} & \sum_{e \in C_{i,j}} x_e \geq |C_{i,j}| y_{i,j}, \ 1 \leq i \leq k, 1 \leq k \leq l_i \\ & \sum_{j=1}^{l_i} y_{i,j} \geq 1, \ 1 \leq i \leq k \\ & \forall e \in E', \ X_e \in \{0,1\} \\ & 1 \leq i \leq k, 1 \leq j \leq l_i, y_{i,j} \in \{0,1\} \end{aligned}$$

6.3 2×3 grid graphs

In this section we present a minimum cardinality feasible chords addition list for 2×3 grid graphs.

Lemma 6.23. Let $G = \langle U, E \rangle$ be a 2×1 grid graph with a clique on $s_{0,0}, s_{1,0}, s_{2,0}$ and a clique on $s_{0,1}, s_{1,1}, s_{2,1}$. The chords addition list $AL = \{(s_{0,1}, s_{1,0}), (s_{0,1}, s_{2,0}), (s_{1,0}, s_{2,1})\}$ is a feasible chords addition list (see Figure 66).

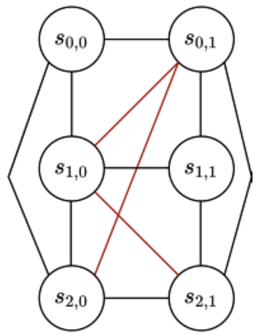


Figure 66: A 2×1 grid graph with a clique on $s_{0,0}, s_{1,0}, s_{2,0}$ and a clique on $s_{0,1}, s_{1,1}, s_{2,1}$. The edges of G are colored black and the edges in AL are colored red.

Proof. G contains the following simple cycles: $C_1 = s_{0,0} - s_{0,1} - s_{2,1} - s_{2,0} - s_{0,0}$, $C_2 = s_{0,0} - s_{0,1} - s_{1,1} - s_{1,0} - s_{0,0}$, $C_3 = s_{1,0} - s_{1,1} - s_{2,1} - s_{2,0} - s_{1,0}$, see Figure 66. AL adds chords to these cycles, achieving chordality on all these cycles, without creating new cycles. Therefore, $G + AL$ is chordal. □

Lemma 6.24. Let $G = \langle U, E \rangle$ be a 2×3 grid graph. Let

$$AL = \{\, (s_{0,1}, s_{1,0}), (s_{0,1}, s_{2,1}), (s_{1,0}, s_{2,1}), (s_{0,2}, s_{1,1}), (s_{0,2}, s_{2,1}), (s_{1,1}, s_{2,2}), \\ (s_{0,2}, s_{1,3}), (s_{0,2}, s_{2,2}), (s_{1,3}, s_{2,2}) \,\}$$

AL is a feasible chords addition list.

Proof. Figure 67 presents $G + AL$, where the edges of G are colored black and the edges of AL are colored red and green.

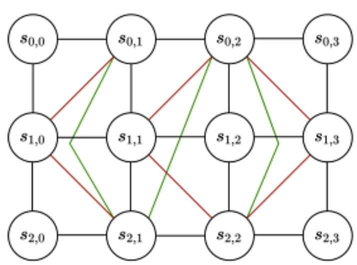


Figure 67: A 2×3 grid graph

The chords $(s_{0,1}, s_{2,1}), (s_{0,2}, s_{1,2})$ slice the graph into three separate subgraphs:

- $U_1 = \{s_{0,0}, s_{1,0}, s_{2,0}, s_{0,1}, s_{1,1}, s_{2,1}\}, G_1 = G[U_1]$.
- $U_2 = \{s_{0,1}, s_{1,1}, s_{2,1}, s_{0,2}, s_{1,2}, s_{2,2}\}, G_2 = G[U_2]$.
- $U_3 = \{s_{0,2}, s_{1,2}, s_{2,2}, s_{0,3}, s_{1,3}, s_{2,3}\}, G_3 = G[U_3]$.

Denote $AL_1 = \{(s_{0,1}, s_{1,0}), (s_{1,0}, s_{2,1})\}$. $G_1 + \{(s_{0,1}, s_{2,1})\}$ is a 2×1 one sided clique grid graph. According to Lemma 6.8, AL_1 is a feasible chords addition list for $G_1 + \{(s_{0,1}, s_{2,1})\}$.

Denote $AL_2 = \{(s_{0,2}, s_{1,1}), (s_{0,2}, s_{2,1}), (s_{1,1}, s_{2,2})\}$. $G_2 + \{(s_{0,1}, s_{2,1}), (s_{0,2}, s_{2,2})\}$ is a 2×1 grid graph with a clique on $s_{0,1}, s_{1,1}, s_{2,1}$ and a clique on $s_{0,2}, s_{1,2}, s_{2,2}$ According to Lemma 6.23, AL_2 is a feasible chords addition list for $G_2 + \{(s_{0,1}, s_{2,1}), (s_{0,2}, s_{2,2})\}$.

Denote $AL_3 = \{(s_{0,2}, s_{1,3}), (s_{1,3}, s_{2,2})\}$. $G_3 + \{(s_{0,2}, s_{2,2})\}$ is a 2×1 one sided clique grid graph. According to Lemma 6.8, AL_3 is a feasible chords addition list for $G_3 + \{(s_{0,2}, s_{2,2})\}$.

Denote $K_1 = G + AL[\{s_{0,1}, s_{1,1}, s_{2,1}\}], K_2 = G + AL[\{s_{0,2}, s_{1,2}, s_{2,2}\}]$ which are cliques in $G + AL$.

Assume by contradiction that $G + AL$ contains a cycle C. C can reside in exactly one, two or three of the subgraphs G_1, G_2, G_3. As shown above, a cycle in one slice cannot exist in exactly one subgraph. If a cycle exists in two or three slices, it must have two nodes in at least one of K_1 or K_2 and in this case one the edges in the clique is a chord in C.

Therefore, $G + AL$ is chordal. □

We will now present another proof for the lemma which uses the connection between CFC and CST.

A second proof for Lemma 6.24. Consider $H = \langle V, \mathcal{S} \rangle$ with $\mathcal{S} = \{S_{0,0}, \ldots, S_{2,0}, S_{0,1}, \ldots, S_{2,1}, S_{0,2}, \ldots, S_{2,2}, S_{0,3}, \ldots, S_{2,3}\}$ and the clusters

$$\begin{aligned}\mathcal{S} = \{\, & S_{0,0} = \{1,2\}, S_{0,1} = \{1,3,4\}, S_{0,2} = \{3,5,6\}, S_{0,3} = \{5,7\},\\ & S_{1,0} = \{2,8,9\}, S_{1,1} = \{4,8,10,11\}, S_{1,2} = \{6,10,12,13\}, S_{1,3} = \{7,12,14\},\\ & S_{2,0} = \{9,15\}, S_{2,1} = \{11,15,16\}, S_{2,2} = \{13,16,17\}, S_{2,3} = \{14,17\}\,\}\end{aligned}$$

We construct a vertices insertion list using external Helly vertices. Let IL be

$$\begin{aligned}IL = \{\, & (30, S_{0,0}),\\ & (30, S_{0,1}), (34, S_{0,1}), (35, S_{0,1}), (36, S_{0,1}),\\ & (32, S_{0,2}), (34, S_{0,2}), (37, S_{0,2}), (38, S_{0,2}),\\ & (32, S_{0,3}),\\ & (30, S_{1,0}), (31, S_{1,0}), (35, S_{1,0}),\\ & (35, S_{1,1}), (36, S_{1,1}),\\ & (34, S_{1,2}), (36, S_{1,2}), (37, S_{1,2}), (38, S_{1,2})\\ & (32, S_{1,3}), (33, S_{1,3}), (38, S_{1,3})\\ & (31, S_{2,0}),\\ & (31, S_{2,1}), (34, S_{2,1}), (35, S_{2,1}), (36, S_{2,1}), (37, S_{2,1}),\\ & (33, S_{2,2}), (37, S_{2,2}), (38, S_{2,2}),\\ & (33, S_{2,3})\,\}\end{aligned}$$

The following is $\mathcal{S} + IL$ where the vertices insertion by IL are colored red. All vertices inserted by

IL are external Helly vertices.

$$\begin{aligned}\mathcal{S}+IL = \{\, & S_{0,0}=\{1,2,30\}, S_{0,1}=\{1,3,4,30,34,35,36\},\\ & S_{0,2}=\{3,5,6,32,34,37,38\}, S_{0,3}=\{5,7,32\},\\ & S_{1,0}=\{2,8,9,30,31,35\}, S_{1,1}=\{4,8,10,11,35,36\},\\ & S_{1,2}=\{6,10,12,13,34,36,37,38\}, S_{1,3}=\{7,12,14,32,33,38\},\\ & S_{2,0}=\{9,15,31\}, S_{2,1}=\{11,15,16,31,34,35,36,37\},\\ & S_{2,2}=\{13,16,17,33,37,38\}, S_{2,3}=\{14,17,33\}\,\}\end{aligned}$$

$G_{int}(\mathcal{S})$ is a 2×3 grid graph and $G_{int}(\mathcal{S}+IL)$ is the graph presented in Figure 67 which is $G+AL$. IL is a feasible vertices insertion list of H, and a feasible solution tree is presented in Figure 68.

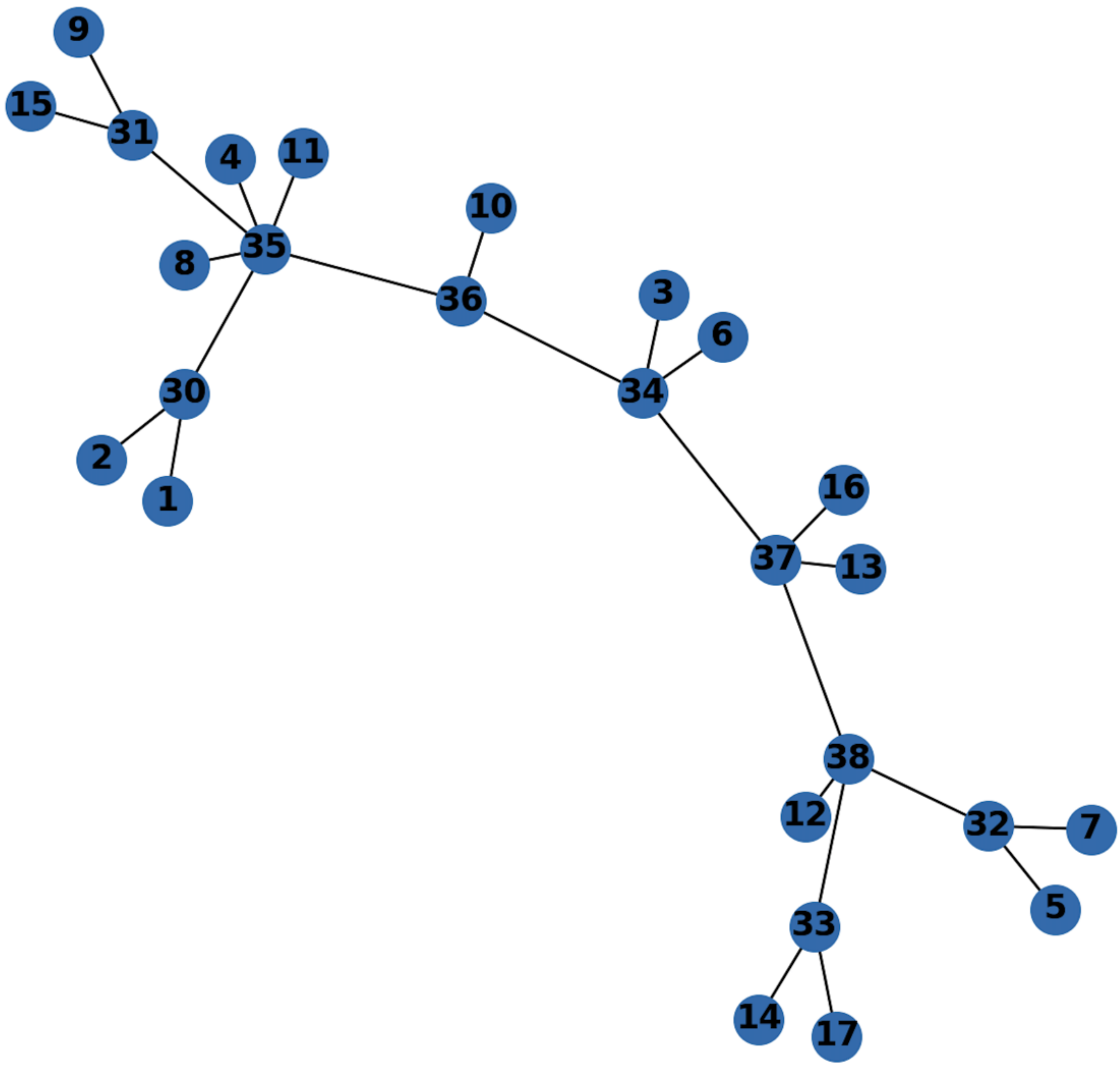


Figure 68: A solution spanning tree for a 2×3 grid hypergraph

According to Theorem 3.2, $G_{int}(\mathcal{S} + IL)$ is chordal and therefore, $G + AL$ is chordal and AL is a feasible chords addition list. $\square$

Lemma 6.25. Let $G = \langle U, E \rangle$ be a 2×3 grid graph. $mAL(G) = 9$.

Proof. G contains the following cycles:

- $C_1 = s_{0,1} - s_{1,2} - s_{1,2} - s_{1,1} - s_{0,1}$, a basic cycle.
- $C_2 = s_{1,1} - s_{1,2} - s_{2,2} - s_{2,1} - s_{1,1}$, a basic cycle.
- $C_3 = s_{0,0} - s_{0,1} - s_{0,2} - s_{0,3} - s_{1,3} - s_{2,3} - s_{2,2} - s_{2,1} - s_{2,0} - s_{1,0} - s_{0,0}$, a ten node cycle.

According to Theorem 6.1, since C_3 is a ten node cycle it requires at least seven chords. In addition, according to Theorem 6.1, since each one of the cycles C_1 and C_2 contains four nodes, they require at least one chord each. The chord added in C_1 cannot be a chord of C_2 and vice versa. In addition, the chords added to C_1 and C_2 cannot be chords of C_3. Therefore, the size of each feasible chords addition list is at least $7 + 1 + 1 = 9$. Lemma 6.24 presents a feasible chords addition list with cardinality nine, and therefore $mAL(G) = 9$. □

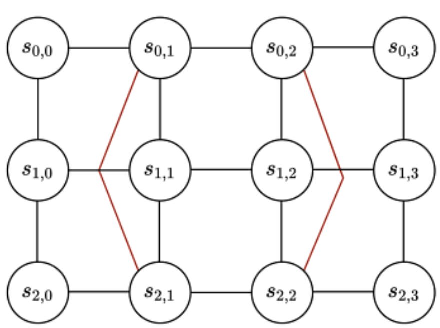
6.3.1 Vertical and Horizontal Slicing of Grids

In this section we present interesting results regarding the slicing of intersection grid graphs for achieving a minimum cardinality feasible chords addition list.

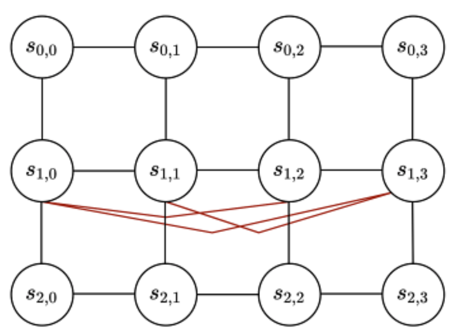
Consider $G = \langle U, E\rangle$ be a 2×3 grid graph. Let $AL = \{(s_{1,0}, s_{0,1}), (s_{1,0}, s_{2,1}), (s_{0,1}, s_{1,2}), (s_{2,1}, s_{1,2}), (s_{0,2}, s_{1,3}), (s_{2,2}, s_{1,3}), (s_{0,1}, s_{2,1}), (s_{0,2}, s_{2,2}), (s_{2,1}, s_{0,2})\}$, $|AL| = 9$. According to Lemma 6.24, AL is a feasible chords addition list which slices the graph into three vertical subgraphs: $U_1 = \{s_{0,0}, s_{1,0}, s_{2,0}, s_{0,1}, s_{1,1}, s_{2,1}\}, G_1 = G[U_1]$, $U_2 = \{s_{0,1}, s_{1,1}, s_{2,1}, s_{0,2}, s_{1,2}, s_{2,2}\}, G_2 = G[U_2]$, $U_3 = \{s_{0,2}, s_{1,2}, s_{2,2}, s_{0,3}, s_{1,3}, s_{2,3}\}, G_3 = G[U_3]$, see Figure 69a.

We may also try to slice the graph horizontally by denoting $U_{up} = \{s_{0,0}, \ldots, s_{0,3}, s_{1,0}, \ldots, s_{1,3}\}$, $G_{up} = G[U_{up}]$, $U_{down} = \{s_{1,0}, \ldots, s_{1,3}, s_{2,0}, \ldots, s_{2,3}\}, G_{down} = G[U_{down}]$, see Figure 69b. Next, we add chords $(s_{1,0}, s_{1,2}), (s_{1,0}, s_{1,3}), (s_{1,1}, s_{1,3})$ to achieve a clique on $U_{up} \cap U_{down} = \{s_{1,0}, s_{1,1}, s_{1,2}, s_{1,3}\}$. After this addition, each of the subgraphs G_{up} and G_{down} becomes a 3×1 one sided clique graph. According to Lemma 6.14, $mAL(G_{up}) = mAL(G_{down}) = 4$. Thus, the horizontal slicing requires $3 + 4 + 4 = 11$ chords, more than the vertical slicing.

Hence, when trying to find the minimum chords addition list of a graph G by slicing, it is important to choose correctly the direction of the slicing.



(a) A 2×3 grid hypergraph sliced vertically



(b) A 2×3 grid hypergraph sliced horizontally

Figure 69: A 2×3 grid hypergraph sliced vertically and horizontally

7 Summary and Further Research

Given a hypergraph, the research focuses on intersection graphs with special characteristics, where it is easy to show that there is no feasible solution for the given hypergraph.

In this research, the first part of the research focuses on CST problem, where the intersection graphs are $n \times m$ grid graphs and $n \times 1$ one sided clique grid graphs. The research starts by looking at a simple 1×1 intersection grid graph scaling up to $n \times 1$ grid graphs, providing a minimum cardinality feasible vertices insertion list. It also provides Convert to Clique method, which enables to easily construct a feasible vertices insertion list for any graph. Then, we prove the cardinality of a minimum cardinality feasible vertices insertion list for $n \times 2$ grid graph.

The second part of the research considers CFC problem, by providing a minimum cardinality feasible chords addition list for 2×1, 3×1 and 4×1 one sided clique grid graphs. In addition, the research provides a method for constructing a feasible chords addition list for an $n \times 1$ one sided clique grid graph, where n is even. The research also uses linear programming to express the minimum cardinality of a chords addition list.

As for further research, we are seeking to generalize the results of this research. For CST problem, it would be interesting to generalize the results for hypergraphs with more complex intersection graphs. For CFC problem, we would like to extend out results including proving minimality for chords addition lists.

References

[1] K. S. Booth and G. S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *Journal of Computer and System Sciences*, 13(3):335–379, December 1976.

[2] P. Duchet. Propriété de Helly et problèmes de représentation. *Colloqu. Internat. CNRS, Problemes Conbinatoires et Theorie du Graphs, Orsay, France*, 260:117–118, 1976.

[3] C. Flament. Hypergraphes arborés. *Discrete Math.*, 21(3), 1978.

[4] C. Florescu. Removals and insertions for a feasible clustered tree by paths. Technical report, The Academic College of Tel Aviv Yaffo, 2021.

[5] N. Guttmann-Beck, R. Rozen, and M. Stern. Vertices removal for feasibility of clustered spanning trees. *To appear in Discrete Applied Mathematics*, 2020.

[6] N. Guttmann-Beck, Z. Sorek, and M. Stern. Clustered spanning tree - conditions for feasibility. *Discrete Mathematics and Theoretical Computer Science*, 21(1), 8 2019.

[7] N. Guttmann-Beck and M. Stern. Decomposing the feasibility question of clustered spanning tree. Technical report, The Academic College of Tel Aviv Yaffo, 2020.

[8] A. A. Hagberg, D. A. Schult, and P. J. Swart. Exploring network structure, dynamics, and function using networkx. In Gaël Varoquaux, Travis Vaught, and Jarrod Millman, editors, *Proceedings of the 7th Python in Science Conference*, pages 11 – 15, Pasadena, CA USA, 2008.

[9] A. Levin. Feasibility of clustered spanning trees by trees for domino intersection graphs. Technical report, The Academic College of Tel Aviv Yaffo, 2020.

[10] T. A. McKee and F. R. McMorris. *Topics in Intersection Graph Theory*. SIAM Monographs on Discrete Mathematics and Applications. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1999.

[11] H. Sayag. Vertices removal for a feasible clustered travelling salesman problem. Technical report, The Academic College of Tel Aviv Yaffo, 2020.

[12] P. J. Slater. A characterization of soft hypergraphs. *Canad. Math. Bull.*, 21(3):335–337, 1978.

[13] R. E. Tarjan and M. Yannakakis. Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM Journal on Computing*, 13(3):566–579, January 1984.

A Proving a unique solution for a chords addition list of 4×1 one sided clique grid graph

The following python code implements the finding all permutations for a chords addition list of size six.

```
import networkx as nx

def __add_basic_chord(option, graph, cycle_index):
    left_to_right = True if option == '0' else False

    basic_cycles = []
    basic_chords = {}
    for i in range(4):
        item = {
            'left': 's(%s,0)' % i,
            'right': 's(%s,1)' % i,
        }
        basic_cycles.append(item)
        basic_chords['s(%s,0)' % i] = 's(%s,1)' % (i + 1)
        basic_chords['s(%s,1)' % i] = 's(%s,0)' % (i + 1)

    cycle = basic_cycles[cycle_index]
    if left_to_right:
        graph.add_edge(cycle['left'], basic_chords[cycle['left']])
    else:
        graph.add_edge(cycle['right'], basic_chords[cycle['right']])


def __create_graphs_with_chords_in_basic_cycles(graph_template):
    '''
    Create graphs permutations with chords only in basic cycles.
    The function uses the binary_number from 0000 to 1111.
    For each character, if a character is 0,
    then an edge is added from left to right,
    otherwise an edge is added from right to left.
    '''
    graphs = []
    for i in range(16):
        graph = nx.Graph()
        graph.add_edges_from(graph_template)
        binary_number = format(i, '04b')
        for j in range(4):
            __add_basic_chord(binary_number[j], graph, j)

        graphs.append(graph)

    return graphs
```

```python
def __test_for_chordality(edges_of_graph, extra_edge_1, extra_edge_2):
    graph = nx.Graph()
    graph.add_edges_from(edges_of_graph)
    graph.add_edges_from(extra_edge_1)
    graph.add_edges_from(extra_edge_2)

    if nx.is_chordal(graph):
        print('graph is chordal', list(graph.edges))


def run():
    graph_template = [
        ('s(0,0)', 's(0,1)'), ('s(0,0)', 's(1,0)'),
        ('s(0,1)', 's(1,1)'), ('s(1,0)', 's(1,1)'),
        ('s(1,0)', 's(2,0)'), ('s(1,1)', 's(2,1)'),
        ('s(2,0)', 's(2,1)'), ('s(2,0)', 's(3,0)'),
        ('s(2,1)', 's(3,1)'), ('s(3,0)', 's(3,1)'),
        ('s(3,0)', 's(4,0)'), ('s(3,1)', 's(4,1)'),
        ('s(4,0)', 's(4,1)'), ('s(0,0)', 's(2,0)'),
        ('s(0,0)', 's(3,0)'), ('s(0,0)', 's(4,0)'),
        ('s(1,0)', 's(3,0)'), ('s(1,0)', 's(4,0)'),
        ('s(2,0)', 's(4,0)'),
    ]
    all_graphs = __create_graphs_with_chords_in_basic_cycles(graph_template)

    for graph in all_graphs:
        edges_in_graph = list(graph.edges)

        print('testing new graph')
        extra_edge_1 = None
        extra_edge_2 = None
        for node_1 in graph:
            for node_2 in graph:
                if node_1 != node_2:
                    extra_edge_1 = [(node_1, node_2)]

                for node_3 in graph:
                    for node_4 in graph:
                        if node_3 != node_4:
                            extra_edge_2 = [(node_3, node_4)]

                        if extra_edge_1 and extra_edge_2:
                            __test_for_chordality(
                                edges_in_graph, extra_edge_1, extra_edge_2
                            )
```

```
                    extra_edge_2 = None

                extra_edge_1 = None

    print('done!')


if __name__ == "__main__":
    run()
```