Removals and Insertions for a Feasible Clustered Tree by Paths

Under the supervision of Dr. Nili Beck and Prof. Michal Stern

Chen Florescu

February 28, 2021

Contents

1 Introduction

Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, where V is a set of vertices and $\mathcal{S}$ is a set of not necessarily disjoint clusters (also known as hyperedges) $S_1, \ldots, S_m$, $S_i \subseteq V$ for $i \in \{1, \ldots, m\}$. The Clustered Spanning Tree by Paths problem, denoted by *CSTP*, is 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.

Since the majority of hypergraphs do not have a feasible solution tree, the question of how to gain feasibility is of great importance. This paper focuses on finding feasible solution trees by removing or inserting a minimum number of vertices from or into the clusters of the given hypergraphs.

The main idea of this paper is to introduce a minimum feasible removal list and a minimum feasible insertion list for a given hypergraph H. A feasible removal (insertion) list contains a list of vertices and clusters, such that removing (inserting) those vertices from (into) the appropriate clusters creates a hypergraph with a feasible solution tree. We consider intersection graph, whose nodes represent the clusters of the hypergraph and an edge exists between two nodes if and only if the corresponding clusters intersect. We focus on cases where the intersection graph has a specific shape, specifically, triangular base shapes, such as a diamond and a butterfly. The research also deals with intersection graphs with special characteristics, where it is easy to show that there is no feasible solution for the given hypergrah. For example, an intersection graph which is a single chordless cycle, or an intersection graph with two chordless cycles connected by a separating edge or a separating path of size three. We also consider cactus tree intersection graph and triangle free intersection graph.

Throughout this paper, we assume that the intersection graph of H is connected. Otherwise, a feasible solution tree of H can be constructed by properly adding edges between the feasible solution trees of each connected component, if they exist. Moreover, when no feasible solution tree exists, the union of feasible minimum removal (insertion) lists of the various connected components, creates a feasible minimum removal (insertion) list for the given hypergraph.

Swaminathan and Wagner in [10] introduce a polynomial time algorithm, which constructs a feasible solution tree for *CSTP* problem, if one exists. Brandes et al. in [2] give a polynomial time algorithm that computes a feasbile solution tree for *CSTP* problem, if it exists. Their algorithm connects subpaths in a specific order using a special coloring of their end vertices.

A generalization of the *CSTP* problem is the Clustered Spanning Tree by Trees problem, denoted by $CSTT$. This problem aims to decide whether there exists a tree-based tree support, which is a tree spanning the vertices of V, such that each cluster induces a subtree. Since *CSTP* is a special case of $CSTT$, obviously the necessary and sufficient conditions presented in [8] for the $CSTT$ problem are necessary conditions also for *CSTP* problem, but not sufficient.

Considering the feasibility question of *CSTP*, in [5] they break the intersection graph of H into smaller instances, when the intersection graph contains a cut node or a separating edge. They prove how the feasibility question of every connected component may be used to decide whether the original hypergraph has a feasible solution tree. In cases where a connected component does not have a feasible solution, they consider changes of the given hypergraph to gain feasibility.

An important known and more restricted version of the *CSTP* problem is where the solution tree is required to be a path, such that every cluster induces a subpath in the solution path. This is the feasibility vertsion of the clasterd *TSP* problem. A solution to this problem can also be presented as 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 1's consecutive in every column. In [1] Booth and Lueker introduce a data structure called a PQ-tree. PQ-trees can be used to represent the permutations of the vertices in *V*, such that the vertices of each cluster of $\mathcal{S}$ are required to occur consecutively.

We would like to suggest a few possible applications for *CSTP* problem. The first one comes from the area of communication networks and is presented by Tanenbaum and Wetherall in [11]. Given a complete graph where each vertex represents a customer, each edge represents a link between two customers, and there is a collection of not necessarily disjoint clusters of vertices where each cluster represents a group of customers. The problem is to construct a communication network in such a way that each cluster of vertices from the given collection induces a path. When using a minimum number of edges, the resulting network is a tree. Note that when no feasible solution tree exists, we consider removing some customers from some of the groups, or inserting some customers into some of the groups, in order to achieve feasibility.

An important use for *CSTP* problem, comes from the area of VLSI design, as is described by Johnson and Pollak in [6]. The vertices of the hypergraph represent electric components and the clusters represent electric subcircuits that should be wired together. The problem is to construct a hypergraph in such a way that each cluster of vertices from the given collection induces a path. For VLSI design it is also of most importance to gain proper hypergraph visualization. Note that when no feasible solution tree exists, we consider removing some components from some of the subcircuits, or inserting some components into some of the subcircuits, in order to achieve feasibility.

This paper is organized as follows: Section 2 describes the connection between the *CSTP* and *CSTT* problems. Section 3 contains definitions that will be used throughout the work. Section 4 contains properties relevant to all the paper. Section 5 deals with triangle intersection graph. Section 6 deals with diamond intersection graph. Section 7 deals with butterfly intersection graph. Section 8 deals with windmill intersection graph. Section 9 deals with vertex connected triangle chain intersection graph. Section 10 deals with edge connected triangle chain intersection graph. Section 11 deals with one chorless cycle intersection graph. Section 12 deals with two chordless cycles with a separating edge intersection graph. Section 13 deals with two chordless cycles with a separating path intersection graph. Section 14 deals with triangular cactus intersection graph. Section 15 deals with cactus intersection graph. Section 16 deals with the triangle free intersection graph.

2 CSTP Versus CSTT

Consider the general following problem: Let $H = < V, S >$ be a hypergraph, where V is a set of vertices and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of not necessarily disjoint clusters, $S_i \subseteq V$, for $1 \leq i \leq m$. The Clustered Spanning Tree by Trees problem, denoted by *CSTT*, is to decide whether there exists a tree spanning the vertices in V, such that each cluster induces a subtree.

Definition 2.1. *A* **chordless cycle** *in a graph is a cycle with at least four vertices, which does not contain any chord. A graph is* **chordal** *when it does not contain any chordless cycle.*

Definition 2.2. *Let $\mathcal{S} = \{S_1, \ldots, S_m\}$ be a family of subsets. $\mathcal{S}$ satisfies the* **Helly Property** *if the following holds: For every $\mathcal{S}' \subseteq \mathcal{S}$, if every pair* *members of $\mathcal{S}'$ has a common element, 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 *CSTP* problem is in fact a restricted case of *CSTT*, as paths are a restricted case of trees. For the *CSTT* problem, it is proved in [3], [4], [9] and summarized in [8], necessary and sufficient conditions for a feasible solution.

Theorem 2.3. *A hypergraph $H = < G, S >$ has a feasible solution tree by trees if and only if H satisfies the Helly property and its intersection graph is chordal.*

Since *CSTP* is a special case of *CSTT*, the above theorem gives necessary conditions for *CSTP*, but not sufficient.

Throughout this work we assume H satisfies the Helly property, otherwise it is clear that H does not have a feasible solution tree by paths.

3 Definitions

In this section we introduce definitions that are used throughout the work.

Definition 3.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S}$ $= \{S_1, \ldots, S_m\}$ a set of not necessarily disjoint clusters. The **intersection graph** of H, denoted by $\mathbf{G_{int}}(\mathcal{S})$, is defined to be a graph whose set of nodes is $\{s_1, \ldots, s_m\}$, where s_i corresponds to S_i, for $i \in \{1, ..., m\}$, and an edge (s_i, s_j) exists if $S_i \bigcap S_j \neq \emptyset$.*

Definition 3.2. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of not necessarily disjoint clusters. Let $\mathcal{S}' \subseteq \mathcal{S}$ be a set of clusters. We define $\mathbf{G}[\mathcal{S}']$ to be the graph whose vertex set is $V(\mathcal{S}') = \bigcup_{S_i \in \mathcal{S}'} S_i$ and cluster set is $\mathcal{S}'$.*

Definition 3.3. *Given a tree T which spans the vertices of V, the subtree of T induced by V', for $V' \subseteq V$, denoted by $\boldsymbol{T[V']}$, is defined to contain all the vertices of V' and all the edges of T whose both endpoints are in V'.*

Definition 3.4. *v^* is a* **cut node** *of a connected graph G if G contains node v^* and deleting v^* from G disconnects G into ξ connected components, for $\xi \geq 2$.*

Definition 3.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S}$ $= \{S_1, S_2, S_3\}$ a set of clusters. A* ***triangular intersection graph*** *of H, is an intersection graph whose nodes set is $\{s_1, s_2, s_3\}$, and its edges set is $\{(s_1, s_2), (s_1, s_3), (s_2, s_3)\}$.*

Definition 3.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S}$ $= \{S_1, S_2, S_3, S_4\}$ a set of clusters. A* ***diamond intersection graph on*** $\mathbf{S_1}, \mathbf{S_2}$ *of H, is an intersection graph whose nodes set is $\{s_1, s_2, s_3, s_4\}$, and its edges set is $\{(s_1, s_2), (s_1, s_3), (s_1, s_4), (s_2, s_3), (s_2, s_4)\}$.*

Definition 3.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of clusters. A* ***butterfly intersection graph on*** $\mathbf{S_1}, \mathbf{S_2}$ *of H, is an intersection graph whose nodes set is $\{s_1, s_2, ..., s_m\}$ and its edges set is $\{(s_1, s_2), (s_1, s_i), (s_2, s_i) \mid i \in \{3, ..., m\}\}$.*

For $i \in \{3, ..., m\}$, a ***wing*** *in a butterfly intersection graph on S_1, S_2, is a sub-graph of the intersection graph, whose nodes set is $\{s_1, s_2, s_i\}$ and its edges set is $\{(s_1, s_2), (s_1, s_i), (s_2, s_i)\}$.*

Definition 3.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of clusters. A* ***windmill intersection graph on*** $\mathbf{S_1}$ *of H, is an intersection graph with $\frac{m-1}{2}$ triangular intersection graphs connected by one node which is s_1.*

Definition 3.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S}$ $= \{S_1, \ldots, S_m\}$ a set of clusters. A* ***vertex connected triangular chain intersection graph*** *of H, is defined to be an intersection graph with $\frac{m-1}{2}$ triangular intersection graphs. Each triangular is connected to its neighbors by one different node.*

Definition 3.10. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S}$ $= \{S_1, \ldots, S_m\}$ a set of clusters. An* ***edge connected triangular chain intersection graph*** *of H, is defined to be an intersection graph with $m-2$ triangular intersection graphs. Each triangular is connected to its neighbors by one different edge.*

Definition 3.11. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S}$ $= \{S_1, \ldots, S_m\}$ a set of clusters. A* **chordless cycle** *is a cycle with at least four nodes, which does not contain any chords.*

Definition 3.12. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of clusters. A* ***two chordless cycles with a***

separating edge intersection graph *of H, is defined to be an intersection graph which contains a separating edge (s_1, s_2), whose removal of nodes $\{s_1, s_2\}$ and edge (s_1, s_2) creates two connected components corresponding to the clusters collections $\mathcal{S}_a, \mathcal{S}_b$. However, the intersection graph remains connected if we remove only one of the vertices s_1 or s_2.*

Definition 3.13. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of clusters. A* ***two chordless cycles with a separating path intersection graph*** *of H, is defined to be an intersection graph which contains a separating path $(s_1, ..., s_t)$, where $t \geq 3$, whose removal of nodes $\{s_1, ..., s_t\}$ and all edges related to these nodes creates two connected components corresponding to the clusters collections $\mathcal{S}_a, \mathcal{S}_b$. However, the intersection graph remains connected if we remove only one of the vertices $s_1, ..., s_t$.*

Definition 3.14. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of clusters. A* ***cactus intersection graph*** *of H, is an intersection graph which is a connected graph in which any two simple chordless cycles have at most one node in common.*

Definition 3.15. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of clusters. A* ***triangular cactus intersection graph*** *of H, is a cactus intersection graph such that each cycle has length three.*

Definition 3.16. *Let $H =< V, \mathcal{S} >$ be a hypergraph with vertex set V and $\mathcal{S} = \{S_1, \ldots, S_m\}$ a set of clusters. A* ***triangle free intersection graph*** *of H, is a graph which does not contain any triangles. Hence, every cycle in this graph contains at least 4 nodes.*

Definition 3.17. *For $\forall\ 1 \leq i \leq m$, let $\mathbf{X_i} = \mathbf{S_i} \setminus \bigcup\{\mathbf{S_r} \mid \mathbf{r} \neq \mathbf{i}, \mathbf{1} \leq \mathbf{r} \leq \mathbf{m}\}$, X_i contains the vertices of S_i that do not appear in any other cluster.*

Definition 3.18. *$\forall\ 1 \leq i, j \leq m$, $j \neq i$, let $\mathbf{X_{i,j}} = (\mathbf{S_i} \bigcap \mathbf{S_j}) \setminus \bigcup\{\mathbf{S_r} \mid \mathbf{r} \neq \mathbf{i}, \mathbf{j}, \mathbf{1} \leq \mathbf{r} \leq \mathbf{m}\}$, $X_{i,j}$ contains the vertices of the intersection of S_i and S_j, that do not appear in any other cluster.*

Definition 3.19. *$\forall\ 1 \leq i, j, k \leq m$, different indices i, j, k, let $\mathbf{X_{i,j,k}} = (\mathbf{S_i} \bigcap \mathbf{S_j} \bigcap \mathbf{S_k}) \setminus \bigcup\{\mathbf{S_r} \mid \mathbf{r} \neq \mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{1} \leq \mathbf{r} \leq \mathbf{m}\}$, $X_{i,j,k}$ contains the vertices of the intersection of S_i, S_j and S_k, that do not appear in any other cluster.*

Definition 3.20. *$\forall\ 1 \leq i,j,k,l \leq m$, different indices i,j,k,l, let* $\mathbf{X_{i,j,k,l}} = (\mathbf{S_i} \bigcap \mathbf{S_j} \bigcap \mathbf{S_k} \bigcap \mathbf{S_l}) \setminus \bigcup \{\mathbf{S_r} \mid \mathbf{r} \neq \mathbf{i}, \mathbf{j}, \mathbf{k}, \mathbf{l}, \mathbf{1} \leq \mathbf{r} \leq \mathbf{m}\}$, *$X_{i,j,k,l}$ contains the vertices of the intersection of S_i, S_j, S_k and S_l, that do not appear in any other cluster.*

Definition 3.21. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_i, S_j, S_k\}$ with a triangular intersection graph. H is a **satisfied triangle on** $\mathbf{S_i}, \mathbf{S_j}$, if at least one of the following holds:*

1. $|X_{i,j,k}| = 1$.
2. $|X_{i,k}| = 0$.
3. $|X_{j,k}| = 0$.

Definition 3.22. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_i, S_j, S_k\}$, with a triangular intersection graph. H is a **strongly satisfied triangle on** $\mathbf{S_i}, \mathbf{S_j}$, if at least one of the following holds :*

1. $|X_{i,j,k}| = 1$.
2. $|X_{i,j}| = 0$.

Definition 3.23. *Let $H =< V, \mathcal{S} >$ be a hypergraph. **RL** is a **removal list** of H if RL is a list of pairs: $RL = \{(v_1, S_{i_1}), ..., (v_k, S_{i_k})\}$ with $v_j \in S_{i_j}$, such that if we remove for all the pairs in RL, vertex v_j from cluster S_{i_j}, we create a new instance of the hypergraph denoted by $\boldsymbol{H} \setminus \boldsymbol{RL}$. If $H \setminus RL$ has a feasible solution tree we say that RL is a **feasible removal list** of H. If RL is also of minimum cardinality (minimum value of k) we say that RL is a **minimum feasible removal list** of H.*

Definition 3.24. *Let $H =< V, \mathcal{S} >$ be a hypergraph. **IL** is an **insertion list** of H if IL is a list of pairs: $IL = \{(v_1, S_{i_1}), ..., (v_k, S_{i_k})\}$ with $v_j \notin S_{i_j}$, such that if we insert for all the pairs in IL, vertex v_j to cluster S_{i_j}, we create a new instance of the hypergraph denoted by $\boldsymbol{H} + \boldsymbol{IL}$. If $H + IL$ has a feasible solution tree we say that IL is a **feasible insertion list** of H. If IL is also of minimum cardinality (minimum value of k) we say that IL is a **minimum feasible insertion list** of H.*

Note that a cluster may appear in the list a few times, each time with a different vertex.

Definition 3.25. *Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph. If $RL = \{(v_1, S_{i_1}), \dots, (v_k, S_{i_k})\}$ is a removal list and $\mathcal{S}' \subseteq \mathcal{S}$ is a set of clusters, we define the* **induced removal list** $\boldsymbol{RL[\mathcal{S}']}$ *to be $\{(v, S_i) \mid (v, S_i) \in RL, S_i \in \mathcal{S}'\}$. Denote by* $\boldsymbol{RL[S_i]}$*$= \{v \mid (v, S_i) \in RL\}$, the vertices removed from S_i by RL.*

Definition 3.26. *Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph. If $IL = \{(v_1, S_{i_1}), \dots, (v_k, S_{i_k})\}$ is an insertion list and $\mathcal{S}' \subseteq \mathcal{S}$ is a set of clusters, we define the* **induced insertion list** $\boldsymbol{IL[\mathcal{S}']}$ *to be $\{(v, S_i) \mid (v, S_i) \in IL, v \in V, S_i \in \mathcal{S}'\}$. Denote by* $\boldsymbol{IL[S_i]}$*$= \{v \mid (v, S_i) \in IL, v \in V\}$, the vertices inserted into S_i by IL.*

Definition 3.27. *Let $H = < V, \mathcal{S} >$ be a hypergraph, with vertex set V and $\mathcal{S} = \{S_1, \dots, S_m\}$ a set of clusters. Let $i, j, k \in \{1, ..., m\}$ be three different indices. Choose $v^* \in X_{i,j,k}$ and let $\mathbf{RL_{i,j,k}} = \{(\mathbf{v}, \mathbf{S_i}) | \mathbf{v} \in \mathbf{X_{i,j,k}}, \mathbf{v} \neq \mathbf{v}^*\}$, a removal list that removes all vertices from $X_{i,j,k}$ except for v^*. After removing $RL_{i,j,k}$ from H, $|X_{i,j,k}| = 1$.*

Definition 3.28. *Let $H = < V, \mathcal{S} >$ be a hypergraph, with vertex set V and $\mathcal{S} = \{S_1, \dots, S_m\}$ a set of clusters. Let $i, j \in \{1, ..., m\}$ be two different indices. Denote $\mathbf{RL_{i,j}} = \{(\mathbf{v}, \mathbf{S_i}) | \mathbf{v} \in \mathbf{X_{i,j}}\}$, a removal list that removes all vertices from $X_{i,j}$. After removing $RL_{i,j}$ from H, $|X_{i,j}| = 0$.*

Definition 3.29. *Let $H = < V, \mathcal{S} >$ be a hypergraph, with vertex set V and $\mathcal{S} = \{S_1, \dots, S_m\}$ a set of clusters. Let $i, j, k \in \{1, ..., m\}$ be three different indices. Denote $\mathbf{IL_{(i,j)+k}} = \{(\mathbf{v}, \mathbf{S_k}) | \mathbf{v} \in \mathbf{X_{i,j}}\}$, an insertion list that inserts all vertices of $X_{i,j}$ to S_k. After inserting $IL_{(i,j)+k}$ to H, $|X_{i,j}| = 0$.*

Definition 3.30. *Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, define* $\boldsymbol{mRL(H)}$*= min $\{|RL| \mid RL$ is a feasible removal list $\}$, the minimum cardinality of all feasible removal lists.*

Definition 3.31. *Let $H = \langle V, \mathcal{S} \rangle$ be a hypergraph, define* $\boldsymbol{mIL(H)}$ *= min $\{|IL| \mid IL$ is a feasible insertion list $\}$, the minimum cardinality of all feasible insertion lists.*

Definition 3.32. *Let* $\mathbf{X^-}$ *be the list of vertices of X after the removal by a removal list.*

Definition 3.33. *Let* $\mathbf{X^+}$ *be the list of vertices of X after the insertion by an insertion list.*

4 General Properties

In this section we introduce general properties that are used throughout the work.

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

Proof. Let $X = \bigcap_{j=1}^{k} S_{i_j}$, where $S_{i_j} \in \mathcal{S}$, and let $\{v, u\} \subseteq X$. It follows that $\{v, u\} \subseteq S_{i_j}\ \forall j \in \{1, \ldots, k\}$. Since T is a feasible solution tree for *CSTP* problem, T contains a path between v and u, such that all the vertices in the path are in S_{i_j}. Therefore, T contains a path between v and u, such that all the vertices in this path are in X. Hence, $T[X]$ is connected and therefore it is a connected subtree of T. Furthermore, since T is a feasible solution tree for *CSTP* problem, $T[S_{i_1}]$ is a path which contains $T[X]$, and therefore $T[X]$ is a connected path. □

Lemma 4.2. *([5]) Consider a hypergraph $H =< V, \mathcal{S} >$ with a connected intersection graph $G_{int}(\mathcal{S})$ and T a feasible solution tree. If $G_{int}(\mathcal{S}')$ is connected for $\mathcal{S}' \subseteq \mathcal{S}$, then $T[V(\mathcal{S}')]$ is a feasible solution tree of $H[\mathcal{S}']$.*

Remark 4.3. *([5]) Consider a hypergraph $H =< V, \mathcal{S} >$ with a connected intersection graph $G_{int}(\mathcal{S})$ and T a feasible solution tree. If $G_{int}(\mathcal{S}')$ is not connected, for $\mathcal{S}' \subsetneq \mathcal{S}$, then according to Theorem 4.2, T induces a feasible solution tree on every connected component of $G_{int}(\mathcal{S}')$, and by adding edges connecting these trees into a tree, a feasible solution tree of $H[\mathcal{S}']$ is achieved.*

Lemma 4.4. *([5]) Consider a hypergraph $H =< V, \mathcal{S} >$. If RL is a feasible removal list for H and $G_{int}(\mathcal{S}')$ is connected, for $\mathcal{S}' \subseteq \mathcal{S}$, then $RL[\mathcal{S}']$ is a feasible removal list for $H[\mathcal{S}']$.*

Lemma 4.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with vertex set V and clusters set $\mathcal{S} = \{S_1, \ldots, S_m\}$. If H has a feasible solution tree by paths, let T be a solution tree. Let $\mathcal{X}', \mathcal{X}'', \mathcal{X}'''$ be sets of intersections of clusters. Let P', P'' and P''' be paths in T which span the intersections $\mathcal{X}', \mathcal{X}'', \mathcal{X}'''$, respectively. If there exists $S_i \in (\mathcal{X}' \cap \mathcal{X}'') \setminus \mathcal{X}'''$, then in any feasible solution P''' can not appear between P' and P''.*

Proof. Suppose by contradiction that H has a feasible solution tree by paths, and P''' is connected between P' and P''. In this case, $P[S_i]$ is not connected, in contradiction with the assumption that H has a feasible solution tree by paths.

□

Lemma 4.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with vertex set V and clusters set $\mathcal{S} = \{S_1, \ldots, S_m\}$. Let T be a feasible solution tree of H. Let $\mathcal{X}', \mathcal{X}'', \mathcal{X}'''$ be sets of intersections of clusters. Let P', P'' and P''' be paths which span the intersections of $\mathcal{X}', \mathcal{X}'', \mathcal{X}'''$, respectively. If there is $S_i \in (\mathcal{X}' \cap \mathcal{X}'' \cap \mathcal{X}''')$, then in any feasible solution there is no vertex $v \in P' \cap P'' \cap P'''$.*

Proof. Suppose by contradiction, that H has a feasible solution tree by paths and there is a vertex $v \in P' \cap P'' \cap P'''$. In this case, P', P'' and P''' all span S_i. Therefore, all tree paths create a tree merging from vertex v, so that $T[S_i]$ is spanned by a tree and not a path, in contradiction with the assumption that H has a feasible solution tree by paths, see Figure 1. □

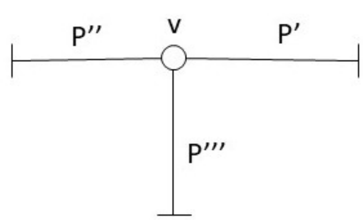

Figure 1: A drawing for Lemma 4.6

Lemma 4.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with vertex set V and clusters set $\mathcal{S} = \{S_1, \ldots, S_m\}$. Let $\mathcal{X}', \mathcal{X}'', \mathcal{X}'''$ be sets of intersections of clusters If H has a feasible solution tree by paths, let P', P'' and P''' be paths which span the intersections of $\mathcal{X}', \mathcal{X}'', \mathcal{X}'''$, respectively. If there exist $S_i \in (\mathcal{X}' \cap \mathcal{X}'') \setminus \mathcal{X}'''$ and $S_j \in (\mathcal{X}' \cap \mathcal{X}''') \setminus \mathcal{X}''$ then in any feasible solution P' has to appear between P'' and P'''.*

Proof. According to Lemma 4.2, P''' can not be connected between P' and P''. Furthermore, P'' can not be connected between P' and P'''. Therefore, the only way to connect the paths is to connect P' between P'' and P'''. In this case, $P[S_i]$ is spanned by the concatenation of P'' and P'. $P[S_j]$ is spanned by the concatenation of P' and P'''. □

Theorem 4.8. *([7]) Let $H =< V, \mathcal{S} >$ be hypergraph whose intersection graph $G_{int}(H)$ is a chordless cycle of size $m \geq 4$, denoted as C, then $|ML(C)| = m - 2$.*

5 Triangular Intersection Graphs

In this section we consider a triangular intersection graph, see Figure 2. We describe the conditions for a *CSTP* solution and suggest a minimum feasible removal list and a minimum feasible insertion list. We assume H satisfies the Helly property, otherwise according to Theorem 2.1, H does not have a feasible solution tree by paths. Thus, by Helly property $|X_{1,2,3}| \geq 1$.

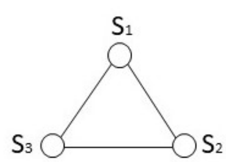

Figure 2: Triangular Intersection Graphs

Theorem 5.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph. If $|X_{1,2,3}| = 1$, then H has a feasible solution tree by paths.*

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 3$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$ and let v be the only vertex in $X_{1,2,3}$. Figure 3 presents a feasible solution by paths for H.

□

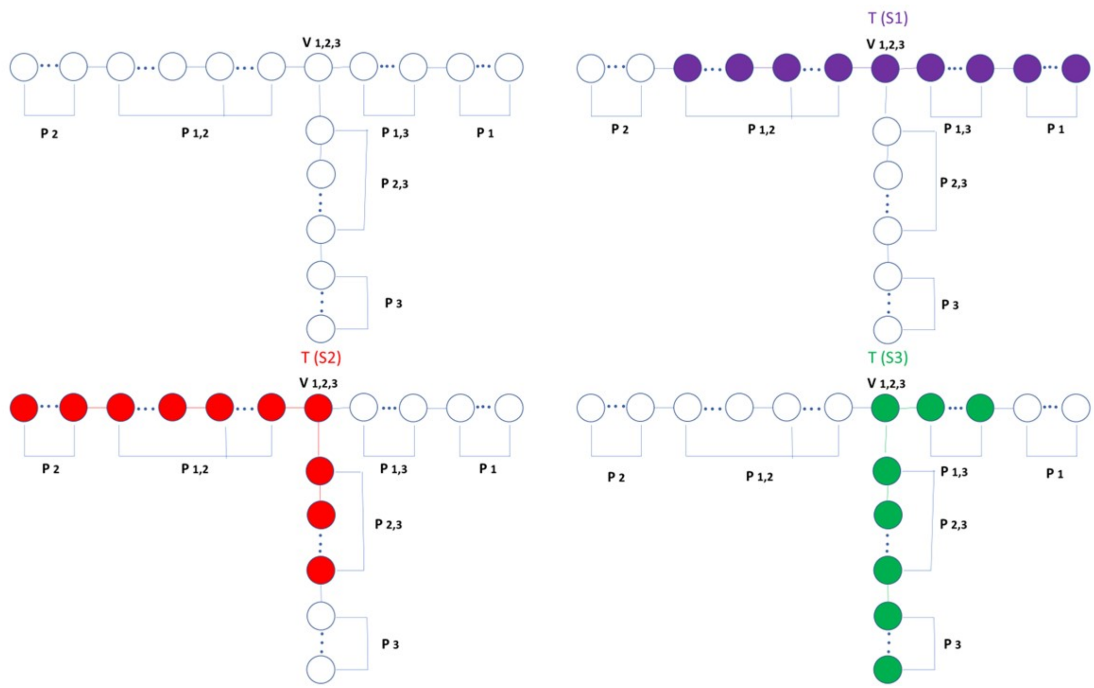

Figure 3: Theorem 5.1 solution tree

Theorem 5.2. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph. If $|X_{1,2,3}| \geq 1$, and at least one of the sub clusters $X_{1,2}$, $X_{1,3}$, $X_{2,3}$ is empty, then H has a feasible solution tree by paths.*

Proof. If $|X_{1,2,3}| = 1$, according to Theorem 5.1, H has a feasible solution tree by paths. Else, without loss of generality, suppose that $|X_{1,2}| = 0$. Figure 4 presents a feasible solution by paths for H. □

Theorem 5.3. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph. If $|X_{1,2,3}| > 1$, and all of the sub clusters $X_{1,2}$, $X_{1,3}$, $X_{2,3}$ are not empty, then H has no feasible solution tree by paths.*

Proof. Suppose by contradiction, that H has a feasible solution tree by paths, denote this tree by T. Since $|X_{1,2,3}| > 1$, according to Lemma 4.1, there is a path $P_{1,2,3}$ with at least one edge which spans $X_{1,2,3}$. According to Lemma 4.1, there is a path $P_{1,2}$ ($P_{2,3}$, $P_{1,3}$) with at least one vertex which spans

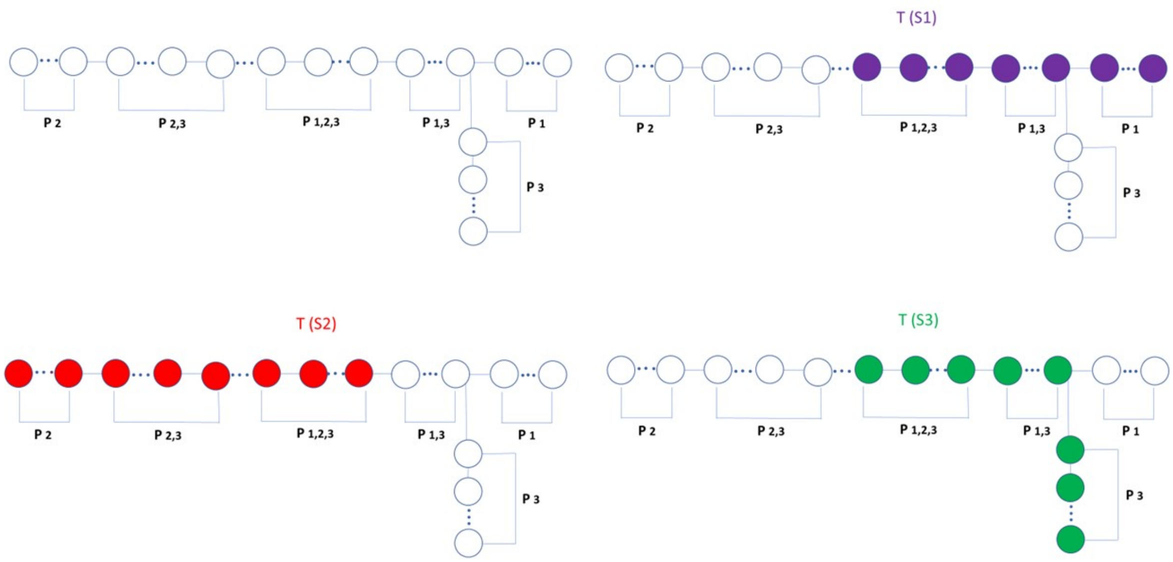

Figure 4: Theorem 5.2 solution tree

$X_{1,2}$ ($X_{2,3}$, $X_{1,3}$) respectively. Since T is a feasible solution tree, $T[S_1]$ is a connected path and therefore it contains $P_{1,2}, P_{1,3}$ and $P_{1,2,3}$ as sub paths. According to Lemma 4.5, $P_{1,2,3}$ has to appear between $P_{1,3}$ and $P_{2,3}$. Furthermore, since T is a feasible solution, $T[S_2]$ is a connected path with $P_{1,2}, P_{2,3}$ and $P_{1,2,3}$ as its sub paths. Thus, according to Lemma 4.5, $P_{1,2,3}$ has to appear between $P_{1,2}$ and $P_{2,3}$.

Hence $P_{1,2,3}$ has to be connected to $P_{1,2}$, $P_{1,3}$ and $P_{2,3}$, such that two of them have to be connected at the same endpoint of $P_{1,2,3}$. Without loss of generality, suppose that $P_{1,2}$ and $P_{1,3}$ are on the same endpoint. However, in this case $T[S_1]$ is spanned by a tree and not a path as shown in Figure 5, contradicting the assumption that T is a feasible solution tree by paths.

□

Corollary 5.4. *According to Theorems 5.1,5.2, 5.3, a triangular intersection graph has a feasible solution tree by paths if and only if* $|X_{1,2,3}|$ *=1 or* $|X_{1,2,3}|$ *>1 and at least one of the sub clusters* $X_{1,2}, X_{1,3}, X_{2,3}$ *is empty.*

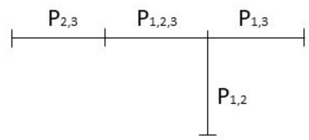

Figure 5: $T[S_1]$

Now we consider removal lists for triangular intersection graphs. Note that, if H has a feasible solution tree, every removal list may be empty.

Theorem 5.5. *Let $H = < V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph. Choose $v^* \in X_{1,2,3}$. $RL_{1,2,3}$ is a feasible removal list of H with cardinality $|X_{1,2,3}| - 1$.*

Proof. Consider $H \setminus RL_{1,2,3}$. According to Definition 3.27, $|X^-_{1,2,3}|$ =1. According to Theorem 5.1, $H \setminus RL_{1,2,3}$ has a feasible solution tree by paths. □

Theorem 5.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph. $RL_{1,2}$ is a feasible removal list with cardinality $|X_{1,2}|$.*

Proof. Consider $H \setminus RL_{1,2}$. According to Definition 3.28, the cardinality of $|X^-_{1,2}| = 0$. Since we assume $|X_{1,2,3}| \geq 1$, according to Theorem 5.2, $H \setminus RL_{1,2}$ has a feasible solution tree by paths. □

Observation 5.7. *Similarly, $RL_{1,3}$ and $RL_{2,3}$ are feasible removal lists with cardinality $|X_{1,3}|$ and $|X_{2,3}|$, respectively.*

Theorem 5.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph.*
$RL \equiv argmin(|RL_{1,2,3}|, |RL_{1,2}|, |RL_{1,3}|, |RL_{2,3}|)$ is a minimum feasible removal list of H.

Proof. According to Theorems 5.5 and 5.6 and Observation 5.7, all the lists in RL are feasible removal lists, therefore RL is a feasible removal list.

Assume H has a feasible solution tree by paths. Then according to Corollary 5.4, one of the lists $RL_{1,2,3}, RL_{1,2}, RL_{1,3}$ or $RL_{2,3}$ is empty, thus by definition RL will also be empty. Therefore, RL is a minimum feasible removal list of H. Otherwise, H does not have a feasible solution tree by paths. According to Corollary 5.4, in order to gain feasibility, either $|X_{1,2,3}|$ =1 or $|X_{1,2,3}|$ >1 and at least one of the sub clusters $X_{1,2}, X_{1,3}, X_{2,3}$ is empty. $RL_{1,2,3}$ represents the first option, $RL_{1,2}, RL_{1,3}$ and $RL_{2,3}$ represent the second option. $RL_{1,2,3}$, $RL_{1,2}, RL_{1,3}$ and $RL_{2,3}$ represent all possible removal lists of H. RL is the list with minimum cardinality, therefore RL is a minimum feasible removal list of H. □

Now we consider insertion lists for triangular intersection graphs. Note that, if H has a feasible solution tree, there is no need for an insertion list.

Theorem 5.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph. $IL_{(1,2)+3}$ is a feasible insertion list of H with cardinality $|X_{1,2}|$.*

Proof. Consider $H + IL_{(1,2)+3}$. According to Definition 3.29, the cardinality of $|X^+_{1,2}| = 0$. According to Theorem 5.2, H has a feasible solution tree by paths. □

Observation 5.10. *Similarly, $IL_{(1,3)+2}$ and $IL_{(2,3)+1}$ are feasible insertion lists with cardinality $|X_{1,3}|$ and $|X_{2,3}|$, respectively.*

Theorem 5.11. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangular intersection graph.*
$IL \equiv argmin(|IL_{(1,2)+3}|, |IL_{(1,3)+2}|, |IL_{(2,3)+1}|)$ is a minimum feasible insertion list of H.

Proof. According to Theorem 5.9 and Observation 5.10, all the lists in IL are feasible insertion lists, therefore IL is a feasible insertion list. Since H has no feasible solution tree by paths, according to Corollary 5.4, $|X_{1,2,3}|$ >1 and all of the sub clusters $X_{1,2}$, $X_{1,3}$, $X_{2,3}$ are not empty. To gain feasibility using insertion, can only be achieved by inserting vertices to $X_{1,2,3}$ and emptying at least one of the sub clusters $X_{1,2}$, $X_{1,3}$ or $X_{2,3}$. According to Theorem 5.9 and Observation 5.10, $IL_{(1,2)+3}$, $IL_{(1,3)+2}$ and $IL_{(2,3)+1}$ represent these insertions and are feasible insertion lists. Therefore $IL \equiv argmin(|IL_{(1,2)+3}|, |IL_{(1,3)+2}|, |IL_{(2,3)+1}|)$ is a minimum feasible insertion list of H. □

5.1 Satisfied Triangles

In this section we consider a satisfied triangle and a strongly satisfied triangle intersection graph. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and a minimum feasible insertion list.

Lemma 5.12. *Let $H =< V, \mathcal{S} >$ be a hypergraph, such that $H[S_1, S_2, S_3]$ is a strongly satisfied triangle on S_1, S_3, then $H[S_1, S_2, S_3]$ has two possible structures for a feasible solution tree by paths.*

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 3$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. If $|X_{1,2,3}| = 1$, let v be the only vertex in $X_{1,2,3}$. Then according to Theorem 5.1, Figure 6.1 presents a feasible solution by paths for H.
If $|X_{1,3}| = 0$. Let $P_{1,2,3}$ be a path spanning $X_{1,2,3}$. Then according to Theorem 5.2, Figure 6.2 presents a feasible solution by paths for H.

□

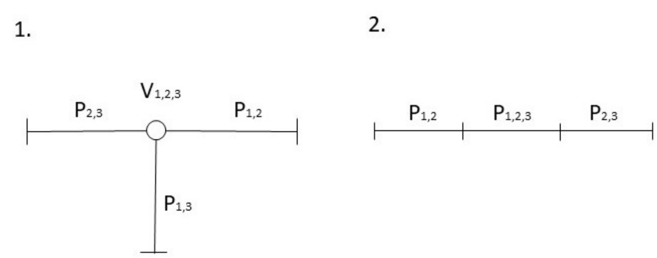

Figure 6: Theorem 5.12 solution trees

Remark 5.13. *Some of the paths in Figure 6, may be empty.*

Lemma 5.14. *Let $H =< V, \mathcal{S} >$ be a hypergraph, such that $H[S_1, S_2, S_3]$ is a satisfied triangle on S_1, S_2, then $H[S_1, S_2, S_3]$ has three possible structures for a feasible solution tree by paths.*

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 3$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$.
If $|X_{1,2,3}| = 1$, let v be the only vertex in $X_{1,2,3}$. Then according to Theorem 5.1, Figure 7.1 presents a feasible solution by paths for H.
If $|X_{2,3}| = 0$, let $P_{1,2,3}$ be a path spanning $X_{1,2,3}$. Then according to Theorem 5.2, Figure 7.2 presents a feasible solution by paths for H.
If $|X_{1,3}| = 0$, let $P_{1,2,3}$ be a path spanning $X_{1,2,3}$. Then according to Theorem 5.2, Figure 7.3 presents a feasible solution by paths for H.

□

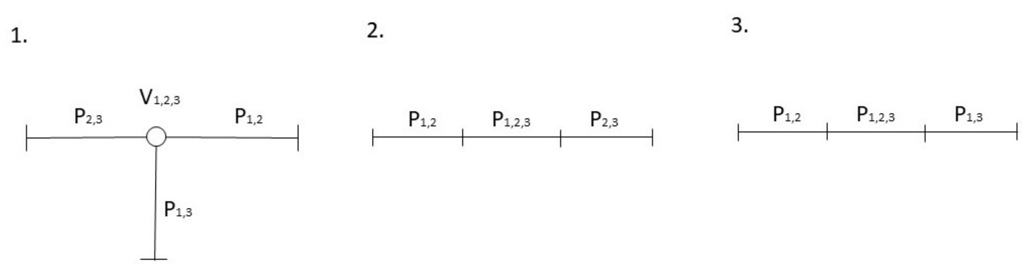

Figure 7: Theorem 5.14 solution trees

Remark 5.15. *Some of the paths in Figure 7, may be empty.*

Now we consider removal lists to gain a satisfied triangular and strongly satisfied triangular intersection graphs.

Theorem 5.16. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangle intersection graph.*
Let $RL_{satisfied} = argmin(|RL_{1,2,3}|, |RL_{1,3}|, |RL_{2,3}|)$.
$RL_{satisfied}$ is a minimum feasible removal list of H, such that $H \setminus RL_{satisfied}$ is a satisfied triangle on S_1, S_2.

Proof. $H[S_1, S_2, S_3]$ is a satisfied triangle on S_1, S_2, if at least one of the following conditions is satisfied: $|X_{1,2,3}| = 1$, $|X_{1,3}| = 0$ or $|X_{2,3}| = 0$.

$RL_{1,2,3}, RL_{1,3}$ and $RL_{2,3}$ represent removal lists to gain each option, respectively. □

Theorem 5.17. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangle intersection graph.*
Let $RL_{strongly} = argmin(|RL_{1,2}|, |RL_{1,2,3}|)$.
$RL_{strongly}$ is a minimum feasible removal list of H, such that $H \setminus RL_{strongly}$ is a strongly satisfied triangle on S_1, S_2.

Proof. $H[S_1, S_2, S_3]$ is a strongly satisfied triangle on S_1, S_2, if at least one of the following conditions is satisfied: $|X_{1,2,3}| = 1$ or $|X_{1,2}| = 0$, $RL_{1,2,3}$ and $RL_{1,2}$ represent removal lists to gain each option, respectively. □

Now we consider insertion lists to gain a satisfied triangular and strongly satisfied triangular intersection graphs.

Theorem 5.18. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangle intersection graph that is not a satisfied triangle on S_1, S_2.*
Let $IL_{satisfied} = argmin(|IL_{(1,3)+2}|, |IL_{(2,3)+1}|)$.
$IL_{satisfied}$ is a minimum feasible insertion list of H, such that $H + IL_{satisfied}$ is a satisfied triangle on S_1, S_2.

Proof. $H[S_1, S_2, S_3]$ is a satisfied triangle on S_1, S_2, if at least one of the following conditions are satisfied: $|X_{1,2,3}| = 1$, $|X_{1,3}| = 0$ or $|X_{2,3}| = 0$. To gain a satisfied triangle on S_1, S_2 using insertions, can only be achieved by inserting vertices to $X_{1,2,3}$ and emptying at least one of the clusters $X_{1,3}$ or $X_{2,3}$. $IL_{(1,3)+2}$ and $IL_{(2,3)+1}$ represent insertion lists, respectively. Thus, $IL_{satisfied}$ is a minimum feasible insertion list of H, such that $H + IL_{satisfied}$ is a satisfied triangle on S_1, S_2. □

Theorem 5.19. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3\}$ and a triangle intersection graph and is not a strongly satisfied triangle on S_1, S_2. Let $IL_{strongly} = IL_{(1,2)+3}$. $IL_{strongly}$ is a minimum feasible insertion list of H, such that $H + IL_{strongly}$ is a strongly satisfied triangle on S_1, S_2.*

Proof. To gain a strongly satisfied triangle on S_1, S_2, one of the following has to hold: $|X_{1,2,3}| = 1$ or $|X_{1,2}| = 0$. To gain a strongly satisfied triangle on S_1, S_2 using insertions, can only be achieved by inserting vertices to $X_{1,2,3}$ and emptying $X_{1,2}$. $IL_{(1,2)+3}$ represents the corresponding insertion list. Thus, $IL_{strongly}$ is a minimum feasible insertion list of H, such that $H + IL_{strongly}$ is a strongly satisfied triangle on S_1, S_2. □

6 Diamond Intersection Graphs

In this section we consider a diamond intersection graph, see Figure 8. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and a minimum feasible insertion list.
We assume H satisfies the Helly property, otherwise according to Theorem 2.1, H does not have a feasible solution tree by paths, therefore $|X_{1,2,3}| \geq 1$ and $|X_{1,2,4}| \geq 1$.
Note that if H has a feasible solution tree, RL will be an empty list.

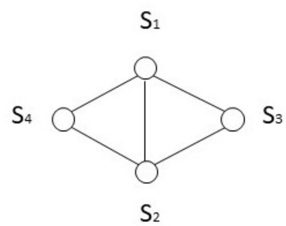

Figure 8: Diamond Intersection Graph

Theorem 6.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ and a diamond intersection graph on S_1, S_2. If $|X_{1,2,3}| = |X_{1,2,4}| = 1$, then H has a feasible solution tree by paths.*

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 4$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. Let $v_{1,2,3}$ be the only vertex in $X_{1,2,3}$. Let $v_{1,2,4}$ be the only vertex in $X_{1,2,4}$. Figure 9 presents a feasible solution by paths for H.

□

Theorem 6.2. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ and a diamond intersection graph on S_1, S_2, with $|X_{1,2,3}| > 1$ and $|X_{1,2,4}| > 1$.*
If $|X_{2,3}| = |X_{2,4}| = 0$, then H has a feasible solution tree by paths.

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 4$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. Let $P_{i,j,r}$ be a path spanning $X_{i,j,r}$, for $i \neq j \neq r$.

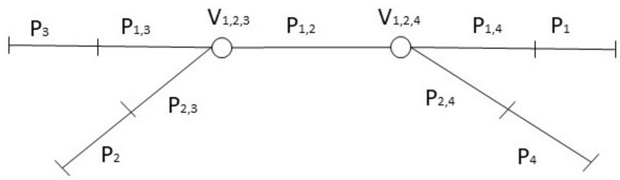

Figure 9: Theorem 6.1 solution tree

Figure 10 presents a feasible solution by paths for H.

□

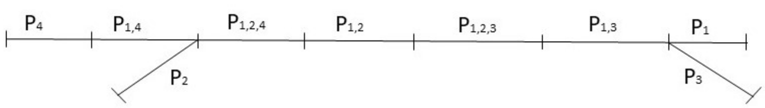

Figure 10: Theorem 6.2 solution tree

Observation 6.3. *Similarly, Theorem 6.2 holds for conditions* $|X_{1,3}| = |X_{1,4}| = 0$*.*

Theorem 6.4. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ *and a diamond intersection graph on* S_1, S_2*, with* $|X_{1,2,3}| > 1$ *and* $|X_{1,2,4}| > 1$*. If* $|X_{2,3}| = |X_{1,4}| = 0$*, then* H *has a feasible solution tree by paths.*

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 4$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. Let $P_{i,j,r}$ be a path spanning $X_{i,j,r}$, for $i \neq j \neq r$.

Figure 11 presents a feasible solution by paths for H.

□

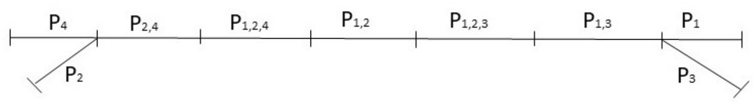

Figure 11: Theorem 6.4 solution tree

Observation 6.5. *Similarly, Theorem 6.4 holds for conditions* $|X_{1,3}| = |X_{2,4}| = 0$.

Remark 6.6. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_i, S_j, S_r, S_k\}$ *and a diamond intersection graph on* S_i, S_j*. Theorem 6.2 is with respect to intersections that share an index* $X_{i,r}$, $X_{i,k}$ *or* $X_{j,r}$, $X_{j,k}$ *. Theorem 6.4 is with respect to intersections which use pairwise disjoint set of indices* $X_{i,r}$, $X_{j,k}$ *or* $X_{j,r}$, $X_{i,k}$.

Theorem 6.7. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ *and a diamond intersection graph on* S_1, S_2*, with* $|X_{1,2,3}| = 1$ *and* $|X_{1,2,4}| > 1$*. If* $|X_{1,4}| = 0$ *or* $|X_{2,4}| = 0$*, then* H *has a feasible solution tree by paths.*

Proof. Without loss of generality, suppose that $|X_{1,4}| = 0$. Let P_i be a path spanning X_i, for $1 \leq i \leq 4$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. Let $P_{i,j,r}$ be a path spanning $X_{i,j,r}$, for $i \neq j \neq r$. Figure 12 presents a feasible solution by paths for H.

□

Observation 6.8. *Similarly, Theorem 6.7 holds for conditions* $|X_{1,2,3}| > 1$, $|X_{1,2,4}| = 1$ *and if* $|X_{1,3}| = 0$ *or* $|X_{2,3}| = 0$.

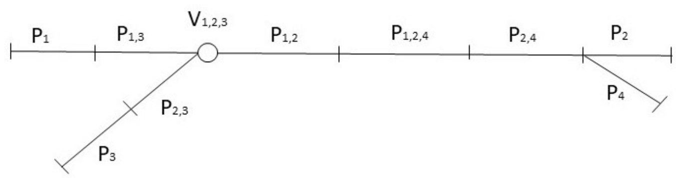

Figure 12: Theorem 6.7 solution tree

Theorem 6.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ and a diamond intersection graph on S_1, S_2, with $|X_{1,2,3}| \geq 1$ and $|X_{1,2,4}| > 1$. If $|X_{1,4}| > 0$ and $|X_{2,4}| > 0$, or $|X_{1,3}| > 0$ and $|X_{2,3}| > 0$, then H has no feasible solution tree by paths.*

Proof. Suppose by contradiction, that H has a feasible solution tree by paths, denote this tree as T. Without loss of generality, suppose that $|X_{1,4}| > 0$ and $|X_{2,4}| > 0$. Since $|X_{1,2,4}| > 1$, according to Lemma 4.1, there is a path $P_{1,2,4}$ with at least one edge which spans $X_{1,2,4}$. Since $|X_{1,2,3}| \geq 1$, there is a path $P_{1,2,3}$ with at least one vertex which spans $X_{1,2,3}$. According to Lemma 4.1, there is a path $P_{1,4}$ ($P_{2,4}$) with at least one vertex which spans $X_{1,4}$ ($X_{2,4}$). Since T is a feasible solution tree, $T[S_2]$ is a connected path which contains $P_{2,4}, P_{1,2,3}$ and $P_{1,2,4}$ as its sub paths. Since T is a feasible solution tree, $T[S_1]$ and $T[S_4]$ are connected, and according to Lemma 4.7, $P_{1,2,4}$ has to be between $P_{1,2,3}$ and $P_{2,4}$. Since T is a feasible solution tree, $T[S_4]$ is connected and contains $P_{1,4}, P_{1,2,4}$ and $P_{2,4}$ as its sub paths, such that $P_{1,2,4}$ is in the middle.
Next we consider how the four sub paths $P_{2,4}, P_{1,2,4}, P_{1,2,3}$ and $P_{1,4}$ are arranged in T. As proven above, $P_{1,2,4}$ is connected between $P_{1,2,3}$ and $P_{2,4}$. $P_{1,4}$ has to be connected to one of the endpoints of $P_{1,2,4}$ to insure that $P[S_1]$, $P[S_2]$ and $P[S_4]$ are connected. Suppose $P_{1,4}$ is connected to the same endpoint as $P_{1,2,3}$. In this case, $P[S_1]$ is spanned by a tree, see Figure 13. Suppose $P_{1,4}$ is connected to the other endpoint of $P_{1,2,4}$ than $P_{1,2,3}$. In this case, $P[S_4]$ is spanned by a tree. Both cases contradict that T is a feasible solution tree by paths.

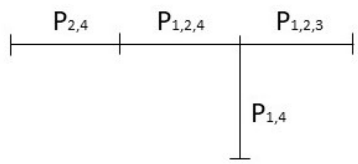

Figure 13: $P[S_1]$ spanned by a tree

□

Observation 6.10. *Similarly, Theorem 6.9 holds for conditions* $|X_{1,2,3}| > 1$, $|X_{1,2,4}| \geq 1$ *and if* $|X_{1,3}| > 0$ *and* $|X_{2,3}| > 0$ *or* $|X_{2,4}| > 0$ *and* $|X_{1,4}| > 0$.

Corollary 6.11. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ *and a diamond intersection graph on* S_1, S_2*.* H *has a feasible solution tree by paths if and only if* $H[S_1, S_2, S_3]$ *and* $H[S_1, S_2, S_4]$ *are satisfied triangles on* S_1, S_2*.*

Proof. Theorems 6.1, 6.2, 6.4 and 6.7 represent all possible ways of $H[S_1, S_2, S_3]$ and $H[S_1, S_2, S_4]$ being satisfied triangles on S_1, S_2 and show a feasible solution tree by paths for H. Therefore, if $H[S_1, S_2, S_3]$ and $H[S_1, S_2, S_4]$ are satisfied triangles on S_1, S_2, H has a feasible solution tree by paths. On the other end, Theorem 6.9 and Observation 6.10, show that if $H[S_1, S_2, S_3]$ or $H[S_1, S_2, S_4]$ are not satisfied triangles on S_1, S_2, then H has no feasible solution tree by paths. □

Now we consider removal lists for diamond intersection graphs. Note that, if H has a feasible solution tree, every removal list may be empty.

Lemma 6.12. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ *and a diamond intersection graph on* S_1, S_2*. The removal of edge* (s_1, s_2) *will not achieve a minimum removal list.*

Proof. The removal of edge (s_1, s_2) can be achieved by removing all of the vertices of $S_1 \cap S_2$ from one of the clusters S_1 or S_2. According to Corollary

6.11, in order to gain a feasible solution tree by paths of H, $H[S_1, S_2, S_3]$ and $H[S_1, S_2, S_4]$ have to be satisfied triangles on S_1, S_2. According to Theorem 5.16, removing edge (s_1, s_2) does not achieve a satisfied triangle on S_1, S_2, therefore this removal will not achieve minimum removal list. □

Theorem 6.13. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ and a diamond intersection graph on S_1, S_2. Let $RL^{1,2,3}$ be a minimum feasible removal list for triangle $H[S_1, S_2, S_3]$, so that $H[S_1, S_2, S_3] \setminus RL^{1,2,3}$ is a satisfied triangle on S_1, S_2. Let $RL^{1,2,4}$ be a minimum feasible removal list for triangle $H[S_1, S_2, S_4]$, so that $H[S_1, S_2, S_4] \setminus RL^{1,2,4}$ is a satisfied triangle on S_1, S_2. $RL \equiv RL^{1,2,3} \bigcup RL^{1,2,4}$ is a minimum feasible removal list of H.*

Proof. By Corollary 6.11, in order to gain a feasible solution tree by paths of H, $H[S_1, S_2, S_3]$ and $H[S_1, S_2, S_4]$ have to be satisfied triangles on S_1, S_2. $H[S_1, S_2, S_3] \setminus RL^{1,2,3}$ is a satisfied triangle on S_1, S_2 and $H[S_1, S_2, S_4] \setminus RL^{1,2,4}$ is a satisfied triangle on S_1, S_2. Another way to achieve feasibility is to remove edge (s_1, s_2), by removing all of the vertices of $S_1 \cap S_2$ from one of the clusters S_1 or S_2. According to Lemma 6.12, this option can never create a minimum removal list.
Therefore, $RL \equiv RL^{1,2,3} \bigcup RL^{1,2,4}$ is a minimum feasible removal list of H. □

Now we consider insertion lists for diamond intersection graphs. Note that, if H has a feasible solution tree, there is no need for an insertion list.

Theorem 6.14. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ and a diamond intersection graph on S_1, S_2. Let $IL^{1,2,3}$ be a minimum feasible insertion list for triangle $H[S_1, S_2, S_3]$, such that $H[S_1, S_2, S_3] + IL^{1,2,3}$ is a satisfied triangle on S_1, S_2. Let $IL^{1,2,4}$ be a minimum feasible insertion list for triangle $H[S_1, S_2, S_4]$, such that $H[S_1, S_2, S_4] + IL^{1,2,4}$ is a satisfied triangle on S_1, S_2. $IL \equiv IL^{1,2,3} \bigcup IL^{1,2,4}$ is a minimum feasible insertion list of H.*

Proof. Proof the same as for Theorem 6.13. □

Lemma 6.15. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4\}$ and a diamond intersection graph on S_1, S_2, with a feasible solution tree by paths of H. Let $P_{1,2}$, $P_{1,2,3}$ and $P_{1,2,4}$ be the paths spanning $X_{1,2}$, $X_{1,2,3}$ and $X_{1,2,4}$, respectively. If $P_{1,2}$ is not connected between $P_{1,2,3}$ and $P_{1,2,4}$, then $P_{1,2}$ can be moved to be connected between $P_{1,2,3}$ and $P_{1,2,4}$, without changing the feasibility of H.*

Proof. H has a feasible solution tree, therefore every cluster in $\mathcal{S}$ is spanned by a connected path. Moving $P_{1,2}$ to be connected between $P_{1,2,3}$ and $P_{1,2,4}$, does not affect the clusters being spanned by a connected path, since $P_{1,2}$ remains in paths $P[S_1]$ and $P[S_2]$. □

7 Butterfly Intersection Graphs

In this section we consider a butterfly intersection graph, see Figure 14. We describe the conditions for a feasible $CSTP$ solution and suggest a minimum feasible removal list and a minimum feasible insertion list.

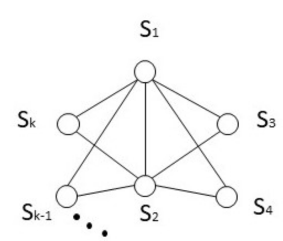

Figure 14: Butterfly Intersection Graph

Theorem 7.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4, S_5\}$ and a butterfly intersection graph on S_1, S_2 with 3 wings. If $H[S_1, S_2, S_3, S_4]$ has a feasible solution and $|X_{1,5}| = |X_{2,5}| = 0$, then H has a feasible solution.*

Proof. According to the theorem's assumption, $H[S_1, S_2, S_3, S_4]$ has a feasible solution tree by paths, denote this tree as T. Since S_1, S_2, S_5 create a wing in the intersection graph, $S_1 \bigcap S_5 \neq \emptyset$ and $S_2 \bigcap S_5 \neq \emptyset$. In addition, since H satisfies the Helly property, $X_{1,2,5} \neq \emptyset$, $X_{1,2,4} \neq \emptyset$ and $X_{1,2,3} \neq \emptyset$.
Let $P_{1,2,4}$ ($P_{1,2,3}$) be the path in T spanning the vertices in $X_{1,2,4}$ ($X_{1,2,3}$). Let $P_{1,2,5}$ be a path spanning $X_{1,2,5}$.
Let $P_{1,2}$ be the path in T spanning the vertices in $X_{1,2}$. If $P_{1,2}$ is not connected between $P_{1,2,3}$ and $P_{1,2,4}$ in T, according to Lemma 6.15, we can change the order of the vertices in $P[S_1]$ such that $P_{1,2}$ is connected between $P_{1,2,3}$ and $P_{1,2,4}$ and T remains a feasible solution tree.

Add $P_{1,2,5}$ between $P_{1,2}$ and $P_{1,2,4}$. Thus, in $P[S_1]$ the sub paths are arranged in the following order: $P_{1,2,3}$, $P_{1,2}$, $P_{1,2,5}$ and $P_{1,2,4}$.
Let P_5 be a path spanning X_5. Recall that $|X_{1,5}| = 0$ and $|X_{1,5}| = 0$, thus connect one of P_5 endpoints to the vertex where $P_{1,2,5}$ and $P_{1,2,4}$ are connected.
S_1 is spanned by $T[S_1]$ and $P_{1,2,5}$. S_2 is spanned by $T[S_2]$ and $P_{1,2,5}$. S_3 is spanned by $T[S_3]$. S_4 is spanned by $T[S_4]$. S_5 is spanned by $P_{1,2,5}$ and P_5. Figure 15 presents a feasible solution by paths for H, for the two case $P_{1,2} \neq \emptyset$ and $P_{1,2} = \emptyset$.

□

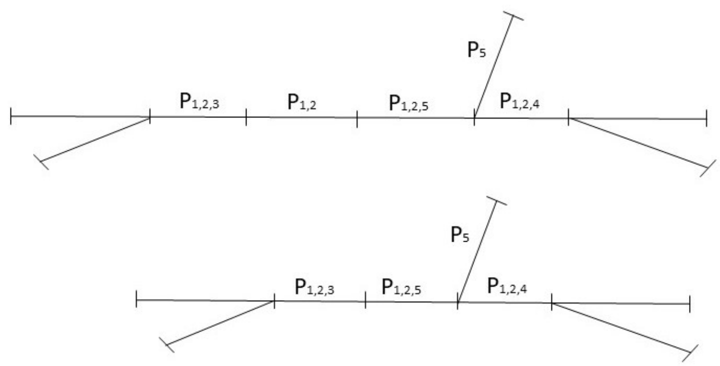

Figure 15: Theorem 7.1 solution tree

Theorem 7.2. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a butterfly intersection graph on S_1, S_2 with k wings. If $H[S_1, S_2, S_3, S_4]$ has a feasible solution and for every $i \in \{5, ..., k\}$ $|X_{1,i}| = |X_{2,i}| = 0$, then H has a feasible solution.*

Proof. Since $G_{int}(H)$ is a butterfly connected intersection graph on S_1, S_2, $H[S_1, S_2, S_3, S_4]$ has a diamond intersection graph on S_1, S_2. According to the theorem's assumption, $H[S_1, S_2, S_3, S_4]$ has a feasible solution tree by paths, denote this tree as T.
Let $P_{1,2,4}$ ($P_{1,2,3}$) be the path in T spanning the vertices in $X_{1,2,4}$ ($X_{1,2,3}$). Let $P_{1,2}$ be the path in T spanning the vertices in $X_{1,2}$.

Since S_1, S_2, S_i where $i \in \{5, ..., k\}$ create a wing in the intersection graph, $S_1 \bigcap S_i \neq \emptyset$ and $S_2 \bigcap S_i \neq \emptyset$. In addition, since H satisfies the Helly property $X_{1,2,i} \neq \emptyset$. Let $P_{1,2,i}$ be the path spanning $X_{1,2,i}$, for $i \in \{5, ..., k\}$.
Concatenate the sub paths $P_{1,2,5}$, $P_{1,2,6}$, ..., $P_{1,2,k}$ in this order. Let P' be the created path. If $P_{1,2}$ is not connected between $P_{1,2,3}$ and $P_{1,2,4}$ in T, according to Lemma 6.15, we can change the order of the vertices in $P[S_1]$ such that $P_{1,2}$ is connected between $P_{1,2,3}$ and $P_{1,2,4}$ and T remains a feasible solution tree.
Connect P' between $P_{1,2}$ and $P_{1,2,4}$.
Let P_i, where $i \in \{5, ..., k-1\}$, be a path spanning X_i, and connect P_i to the vertex connecting $P_{1,2,i}$ and $P_{1,2,i+1}$.
Let P_k be a path spanning X_k, and connect P_k to the vertex connecting $P_{1,2,k}$ and $P_{1,2,4}$. S_1 is spanned by $T[S_1]$ and P' . S_2 is spanned by $T[S_2]$ and P' . S_3 is spanned by $T[S_3]$. S_4 is spanned by $T[S_4]$. S_i is spanned by $P_{1,2,i}$ and P_i. Figure 16 presents a feasible solution by paths for H.

□

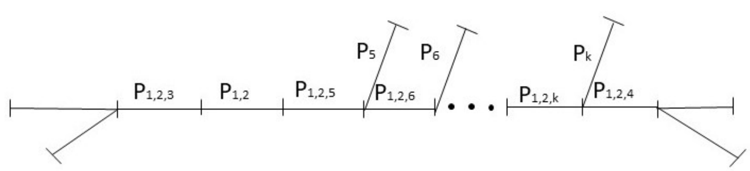

Figure 16: Theorem 7.2 solution tree

Observation 7.3. *Similar to Theorem 7.2, if $H[S_1, S_2, S_i, S_j]$ for $i, j \in \{3, ..., m\}$, $i \neq j$ has a feasible solution and for every $k \in \{3, ..., m\} \setminus \{i, j\}$ $|X_{1,k}| = |X_{2,k}| = 0$, then H has a feasible solution tree.*

Theorem 7.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with a butterfly intersection graph on S_1, S_2. If there are 3 different indices i_1, i_2, i_3 such that $|X_{1,i_j}| > 0$ or $|X_{2,i_j}| > 0$ for $j \in \{1, 2, 3\}$, then H has no feasible solution tree by paths.*

Proof. Suppose by contradiction that H has a feasible solution tree, denoted by T. Without loss of generality, let $\{i_1, i_2, i_3\} = \{3, 4, 5\}$, and that $|X_{1,3}| > 0$, $|X_{1,4}| > 0$, $|X_{1,5}| > 0$.
Since S_1, S_2, S_{i_j} create a wing in the intersection graph, $S_1 \bigcap S_{i_j} \neq \emptyset$ and $S_2 \bigcap S_{i_j} \neq \emptyset$, for $1 \leq j \leq 3$. Since H satisfies the Helly property, $X_{1,2,i_j} \neq \emptyset$. According to Lemma 4.1, every intersection is spanned by a connected path. Let $P_{1,3}, P_{1,4}$ and $P_{1,5}$ be the paths spanning $X_{1,3}$, $X_{1,4}$ and $X_{1,5}$ in T, respectively. Let $P_{1,2,3}, P_{1,2,4}$ and $P_{1,2,5}$ be the paths spanning $X_{1,2,3}$, $X_{1,2,4}$ and $X_{1,2,5}$ in T, respectively. $T[S_1]$ is a connected path with $P_{1,3}, P_{1,4}, P_{1,5}, P_{1,2,3}$, $P_{1,2,5}$ and $P_{1,2,5}$ as its sub paths. According to Lemma 4.7, $P_{1,2,3}$ is between $P_{1,3}$ and $P_{1,2,4}$, and $P_{1,2,4}$ is between $P_{1,4}$ and $P_{1,2,3}$. So the order of the sub paths in $P[S_1]$ is $P_{1,3}, P_{1,2,3}, P_{1,2,4}, P_{1,4}$. According to Lemma 4.5, $P_{1,2,5}$ can not to be connected between $P_{1,3}$ and $P_{1,2,3}$ or between $P_{1,2,4}$ and $P_{1,4}$. Hence the order of the sub paths in $P[S_1]$ is $P_{1,3}, P_{1,2,3}, P_{1,2,5}, P_{1,2,4}, P_{1,4}$.
Next, we consider where the sub path $P_{1,5}$ is inside $P[S_1]$. $P_{1,5}$ has to touch $P_{1,2,5}$ to insure that $P[S_5]$ is connected. However, according to Lemma 4.5, $P_{1,5}$ can not be connected between $P_{1,2,3}$ and $P_{1,2,5}$ or between $P_{1,2,5}$ and $P_{1,2,4}$. Contradicting the assumption that H has a feasible solution tree by paths. □

Now we consider removal lists for butterfly intersection graphs. Note that, if H has a feasible solution tree, every removal list may be empty.

Theorem 7.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a butterfly intersection graph on S_1, S_2. Let $RL^{i,j}$, $i, j \in \{3, ..., m\}$ be a minimum cardinality feasible removal list of $H[S_1, S_2, S_i, S_j]$.*
Let $BRL = RL^{i,j} \bigcup (RL_{1,k} \bigcup RL_{2,k}), k \in \{3, ..., m\} \setminus \{i, j\}$. BRL is a feasible removal list for H.

Proof. Since $RL^{i,j}$ is a feasible removal list, $H[S_1, S_2, S_i, S_j] \setminus RL^{i,j}$ has a feasible solution tree. $H[S_1, S_2, S_i, S_j] \setminus BRL$ has a feasible solution tree. In addition, in $H \setminus BRL$, for every $k \in \{3, ..., m\} \setminus \{i, j\}$, $|X^-_{1,k}| = |X^-_{2,k}| = 0$ and therefore, according to Theorem 7.2, H has a feasible solution tree by paths. □

Algorithm 1: ButterflyMinRemovalList

Input : Butterfly intersection graph
Output: Minimum removal list for butterfly intersection graph
$BRL = [\,]$;
for $i, j \in \{3, ..., m\},\ i \neq j$ **do**
  Find $RL^{i,j}$ a minimum cardinality feasible removal list for $H[S_1, S_2, S_i, S_j]$;
  tempList = [ ];
  for *k such that* $k \in \{3, ..., m\}$ *and* $k \neq i, j$ **do**
    tempList = tempList $\bigcup RL_{1,k} \bigcup RL_{2,k}$;
  end
  $BRL^{i,j} = RL^{i,j} \bigcup tempList$;
  Let $i^*, j^* = argmin(BRL^{i,j})$;
end
return BRL^{i^*,j^*}

Theorem 7.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a butterfly intersection graph on S_1, S_2. Algorithm ButterflyMinRemovalList returns a minimum cardinality feasible removal list for H.*

Proof. Let L be a minimum feasible removal list. According to Theorem 7.4, in $H \setminus L$ there are at most two indices i', j' such that ($|X^-_{1,i'}| > 0$ or $|X^-_{2,i'}| > 0$) and ($|X^-_{1,j'}| > 0$ or $|X^-_{2,j'}| > 0$). Furthermore, in $H \setminus L$, for every $k \in \{3, ..., m\} \setminus \{i', j'\}$, $|X^-_{1,k}| = 0$ and $|X^-_{2,k}| = 0$. According to Theorem 7.5, $H \setminus L[S_1, S_2, S_{i'}, S_{j'}]$ has a feasible solution, therefore $L = RL^{i',j'} \bigcup RL_{1,k} \bigcup RL_{2,k}$ $k \in \{3, ..., m\} \setminus \{i', j'\}$, giving that $L = BRL^{i',j'}$.
Let BRL^{i^*,j^*} be the result of the algorithm ButterflyMinRemovalList. Since algorithm ButterflyMinRemovalList consider all possible pairs of indices, it will also consider i', j', and therefore $|BRL^{i^*,j^*}| \leq |BRL^{i',j'}| = |L|$, giving that BRL^{i^*,j^*} is also a minimum feasible removal list. □

Now we consider insertion lists for the butterfly intersection graph. Note that, if H has a feasible solution tree, there is no need for an insertion list.

Theorem 7.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a butterfly intersection graph on S_1, S_2. Let $IL^{i,j}$, $i, j \in \{3, ..., m\}$ be a minimum cardinality feasible insertion list of $H[S_1, S_2, S_i, S_j]$.*
Let $BIL = IL^{i,j} \bigcup (IL_{(1,k)+2} \bigcup IL_{(2,k)+1}), k \in \{3, ..., m\} \setminus \{i, j\}$. BIL is a feasible insertion list for H.

Proof. Since $IL^{i,j}$ is a feasible insertion list, $H[S_1, S_2, S_i, S_j] + IL^{i,j}$ has a feasible solution tree. $H[S_1, S_2, S_i, S_j] + BIL$ has a feasible solution tree. In addition, in $H + BIL$, for every $k \in \{3, ..., m\} \setminus \{i, j\}$, $|X^+_{1,k}| = |X^+_{2,k}| = 0$ and therefore, according to Theorem 7.2, H has a feasible solution tree by paths. □

Algorithm 2: ButterflyMinInsertionList

Input : Butterfly intersection graph
Output: Minimum insertion list for butterfly intersection graph

$BIL = [\ ]$;
for $i, j \in \{3, ..., m\}$, $i \neq j$ **do**
  Find $IL^{i,j}$ a minimum cardinality feasible insertion list for $H[S_1, S_2, S_i, S_j]$;
  tempList = [ ];
  for k *such that* $k \in \{3, ..., m\}$ *and* $k \neq i, j$ **do**
    tempList = tempList $\bigcup IL_{(1,k)+1} \bigcup IL_{(2,k)+1}$;
  end
  $BIL^{i,j} = IL^{i,j} \bigcup tempList$;
  Let $i^*, j^* = argmin(BIL^{i,j})$;
end
return BIL^{i^*,j^*}

Theorem 7.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a butterfly intersection graph on S_1, S_2. Algorithm ButterflyMinInsertionList returns a minimum cardinality feasible insertion list for H.*

Proof. Let L be a minimum feasible insertion list. According to Theorem 7.4, in $H + L$ there are at most two indices i', j' such that ($|X^+_{1,i'}| > 0$ or $|X^+_{2,i'}| > 0$) and ($|X^+_{1,j'}| > 0$ or $|X^+_{2,j'}| > 0$). Furthermore, in $H + L$ for every $k \in \{3, ..., m\} \setminus \{i', j'\}$, $|X^+_{1,k}| = 0$ and $|X^+_{2,k}| = 0$. According to Theorem 7.7, $H + L[S_1, S_2, S_{i'}, S_{j'}]$ has a feasible solution, therefore $L = IL^{i',j'} \bigcup IL_{(1,k)+1} \bigcup IL_{(2,k)+1}$ $k \in \{3, ..., m\} \setminus \{i', j'\}$, giving that $L = BIL^{i',j'}$. Let BIL^{i^*,j^*} be the result of the algorithm ButterflyMinInsertionList. Since algorithm ButterflyMinInsertionList consider all possible pairs of indices, it will also consider i', j', and therefore $|BIL^{i^*,j^*}| \leq |BIL^{i',j'}| = |L|$, giving that BIL^{i^*,j^*} is also a minimum feasible insertion list. □

8 Windmill Intersection Graphs

In this section we consider a windmill intersection graph, see Figure 17. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and a minimum feasible insertion list.

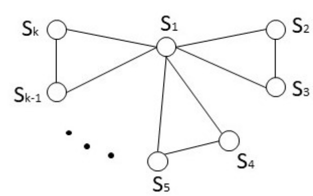

Figure 17: Windmill Intersection Graph

Theorem 8.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a windmill intersection graph on S_1. $G_{int}(H)$ has $\frac{m-1}{2}$ triangular induced sub graphs.*

Proof. H has m clusters, cluster S_1 that corresponds to node s_1 in $G_{int}(H)$, is the cluster connected to all the triangles in $G_{int}(H)$ such that every triangle has 2 more nodes. Therefore, the total number of triangles is $\frac{m-1}{2}$. □

Observation 8.2. *According to Definition 3.8, s_1 is a cut node in $G_{int}(\mathcal{S})$ that disconnects $G_{int}(\mathcal{S})$ into $\frac{m-1}{2}$ connected components whose corresponding cluster sets are $\{S_2, S_3\}, \{S_4, S_5\}, ..., \{S_{m-1}, S_m\}$.*

Observation 8.3. *Furthermore, since s_1 is a cut node, the corresponding vertices sets $S_2 \bigcup S_3, S_4 \bigcup S_5, ..., S_{m-1} \bigcup S_m$ are pairwise vertex disjoint.*

Theorem 8.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a windmill intersection graph on S_1. If every triangle in the windmill has a feasible solution tree by paths, then H has a feasible solution.*

Proof. In [5], they prove that if the connected intersection graph $G_{int}(\mathcal{S})$ contains a cut node s^*, which disconnects the intersection graphs to clusters sets $\{s_a, \ldots, s_\xi\}$ and if every $H_j = H[\mathcal{S}_j \cup \{S^*\}]$, $j \in \{a, \ldots, \xi\}$, has a feasible solution for $CSTP$ problem, then H has a feasible solution for $CSTP$ problem. According to Observation 8.2, and the theorem assumption, every triangle in the windmill has a feasible solution tree by paths. Therefore, H has a feasible solution tree by paths, see Figure 18. □

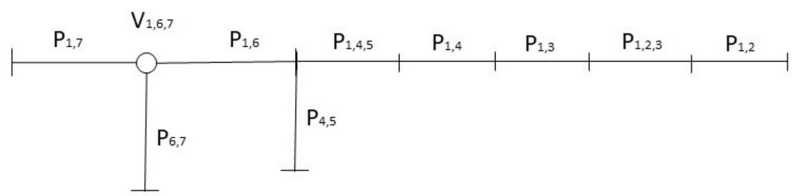

Figure 18: Theorem 8.4 solution tree

Now we consider removal lists for windmill intersection graph. Note that, if H has a feasible solution tree, every removal list may be empty.

Theorem 8.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a windmill intersection graph on S_1. $mRL(H) = \sum_{i=1}^{\frac{m-1}{2}} mRL(H[S_1, S_{2i}, S_{2i+1}])$, $i \in \{1, .., \frac{m-1}{2}\}$.*

Proof. According to Observation 8.2, s_1 is a cut node which divides the intersection graph into $\frac{m-1}{2}$ connected components whose clusters sets are $\{S_1, S_{2i}, S_{2i+1}\}$, for $i \in \{1, .., \frac{m-1}{2}\}$. According to [5], $mRL(H) = \sum_{i=1}^{\frac{m-1}{2}} mRL(H[S_1, S_{2i}, S_{2i+1}])$, $i \in \{1, .., \frac{m-1}{2}\}$. □

Theorem 8.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a windmill intersection graph on S_1. Let RL_i be a minimum feasible removal list for $H[S_1, S_{2i}, S_{2i+1}]$, $i \in \{1, .., \frac{m-1}{2}\}$. $RL \equiv \bigcup_{i=1}^{\frac{m-1}{2}} RL_i$ is a minimum feasible removal list of H.*

Proof. According to Theorem 8.4, if every triangle in the windmill has a feasible solution tree by paths, then H has a feasible solution. Since RL_i is a feasible removal list for $H[S_1, S_{2i}, S_{2i+1}]$, $H[S_1, S_{2i}, S_{2i+1}] \setminus RL_i$ has a feasible solution tree. According to Theorem 8.4, $H \setminus \bigcup_{i=1}^{\frac{m-1}{2}} RL_i$ has a feasible solution tree. Note that if one of the removal lists removes all the vertices from S_1, the intersection graph is disconnected and if every connected component has a solution tree so thus the whole hypergraph. According to Theorem 8.5, $\bigcup_{i=1}^{\frac{m-1}{2}} RL_i$ is also a minimum removal list. □

Now we consider insertion lists for windmill intersection graph. Note that, if H has a feasible solution tree, there is no need for an insertion list.

Theorem 8.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a windmill intersection graph on S_1. $mIL(H) = \sum_{i=1}^{\frac{m-1}{2}} mIL(H[S_1, S_{2i}, S_{2i+1}])$.*

Proof. As shown for Theorem 8.5. □

Theorem 8.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a windmill intersection graph on S_1. Let IL_i be a minimum feasible insertion list for $H[S_1, S_{2i}, S_{2i+1}]$, $i \in \{1, .., \frac{m-1}{2}\}$. $IL \equiv \bigcup_{i=1}^{\frac{m-1}{2}} IL_i$ is a minimum feasible insertion list of H.*

Proof. As shown for Theorem 8.6. □

9 Vertex Connected Triangular Chain Intersection Graphs

In this section we consider a vertex connected triangular chain intersection graph with $\frac{m-1}{2}$ triangular intersection graphs, were each triangular is connected to its neighbors by one different node, see Figure 19. We describe the conditions for a feasible $CSTP$ solution and suggest a minimum feasible removal list and a minimum feasible insertion list.

Observation 9.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a vertex connected triangular chain intersection graph. $H[S_{2i-1}, S_{2i}, S_{2i+1}]$, for $i \in \{1, ..., \frac{m-1}{2}\}$, is a triangular intersection graph.*

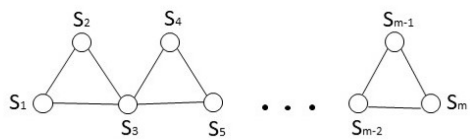

Figure 19: Vertex Connected Chain Intersection Graph

Lemma 9.2. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with a vertex connected triangular chain intersection graph. If the intersection graph has t sub graphs which are triangles, then H has $2t + 1$ clusters.*

Proof. Proof by induction on t, the number of triangular induced sub graphs in H.

If $t = 1$ then $G_{int}(H)$ contains only one triangular with three clusters.

Suppose the claim is correct for $t - 1$. We prove it for t. $G_{int}(H)$ has $t - 1$ triangular induced sub graphs and $2(t - 1) + 1 = 2t - 1$ clusters. We add two clusters to add one more triangular to $G_{int}(H)$, and therefore, $2(t) + 1 = 2t + 1$. □

Observation 9.3. *According to Lemma 9.2, $G_{int}(H)$ has $\frac{m-1}{2}$ triangular induced sub graphs.*

Observation 9.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a vertex connected triangular chain intersection graph. Cluster s_{2i+1} is a cut node that divides the intersection graph to clusters sets $\{S_1, .., S_{2i}\}$ and $\{S_{2i+2}, ..., S_m\}$. Furthermore, $H[S_1, ..., S_{2i+1}]$ and $H[S_{2i+1}, ..., S_m]$ have a vertex connected triangular chain intersection graph, for $i \in \{1, ..., \frac{m-1}{2}\}$.*

Theorem 9.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a vertex connected triangular chain intersection graph. If $H[S_{2i-1}, S_{2i}, S_{2i+1}], i \in \{1, .., \frac{m-1}{2}\}$ has a feasible solution tree by paths, then H has a feasible solution tree by paths.*

Proof. Proof by induction on t, the number of triangular induced sub graphs in $G_{int}(H)$.

If $t = 1$ then $G_{int}(H)$ contains only one triangular. According to the theorem assumption, this triangular has a feasible solution tree by paths. This is a feasible solution tree tree by paths for H.

Suppose the claim is correct for $t-1$. We prove it for t. According to Observation 9.4, s_{2m-1} is a cut node where $H[S_1, ..., S_{2m-1}]$ has a vertex connected chain intersection graph with $t-1$ triangles and $H[S_{2m-1}, S_{2m}, S_{2m+2}]$ has a triangular intersection graph. According to the induction hypothesis, $H[S_1, ..., S_{2m-1}]$ has a feasible solution tree. According to Observation 9.4, s_{2m-1} is a cut node and $H[S_1, ..., S_{2m-1}]$ and $H[S_{2m-1}, S_{2m}, S_{2m+1}]$ have a feasible solution tree by paths. Therefore, according to [5], H has a feasible solution tree tree by paths. □

Now we consider removal lists for vertex connected triangular chain intersection graph. Note that, if H has a feasible solution tree, every removal list may be empty.

Theorem 9.6. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ *and a vertex connected triangular chain intersection graph.* $mRL(H) = \sum_{i=1}^{\frac{m-1}{2}} mRL(H[S_{2i-1}, S_{2i}, S_{2i+1}])$*,* $i \in \{1, .., \frac{m-1}{2}\}$*.*

Proof. Proof that $mRL(H)$ is a minimum removal list by induction on t, the number of triangular induced sub graphs in $G_{int}(H)$.

If $t = 1$ then $G_{int}(H)$ contains only one triangular, according to the theorem assumption, RL contains only the minimum feasible removal list of this triangular. Therefore, $mRL(H)$ is a minimum feasible removal list of H.

Suppose the claim is correct for $t-1$. We prove it for t. According to Theorem 9.5, s_{2t-1} is a cut node where $H[S_1, ..., S_{2t-1}]$ has a vertex connected chain intersection graph with $t-1$ triangles and $H[S_{2t-1}, S_{2t}, S_{2t+1}]$ has a triangular intersection graph. Since s_{2m-1} is a cut node, according to [5], then $mRL(H) = mRL(H[S_1, ..., S_{2t-1}]) + mRL(H[S_{2t-1}, S_{2t}, S_{2t+1}])$. According to the induction hypothesis, $mRL(H[S_1, ..., S_{2t-1}) = \sum_{i=1}^{\frac{t-2}{2}} mRL(H[S_{2i-1}, S_{2i}, S_{2i+1}])$ which proves that, $mRL(H) = \sum_{i=1}^{\frac{m-1}{2}} mRL(H[S_{2i-1}, S_{2i}, S_{2i+1}]), i \in \{1, .., \frac{m-1}{2}\}$. □

Theorem 9.7. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ *and a vertex connected triangular chain intersection graph.*
Let RL_i *be a minimum feasible removal list for* $H[S_{2i-1}, S_{2i}, S_{2i+1}]$*,* $i \in \{1, .., \frac{m-1}{2}\}$*.* $RL \equiv \bigcup_{i=1}^{\frac{m-1}{2}} RL_i$ *is a minimum feasible removal list of* H*.*

Proof. According to Theorem 9.5, if every triangle in the vertex connected chain has a feasible solution tree by paths, then H has a feasible solution. Since RL_i is a feasible removal list for $H[S_{2i-1}, S_{2i}, S_{2i+1}]$, $H[S_{2i-1}, S_{2i}, S_{2i+1}] \setminus RL_i$, for every $i \in \{1, .., \frac{m-1}{2}\}$, has a feasible solution tree. According to Theorem 9.6, $H \setminus \bigcup_{i=1}^{\frac{m-1}{2}} RL_i$ has a feasible solution tree. Hence, $\bigcup_{i=1}^{\frac{m-1}{2}} RL_i$ is a minimum removal list of H. □

Now we consider insertion lists for vertex connected triangular chain intersection graph. Note that, if H has a feasible solution tree, there is no need for an insertion list.

Theorem 9.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a vertex connected triangular chain intersection graph.*
$mIL(H) = \sum_{i=1}^{\frac{m-1}{2}} mIL(H[S_{2i-1}, S_{2i}, S_{2i+1}])$, $i \in \{1, .., \frac{m-1}{2}\}$.

Proof. As shown for Theorem 9.6.

□

Theorem 9.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a vertex connected triangular chain intersection graph.*
Let IL_i be a minimum feasible insertion list for $H[S_{2i-1}, S_{2i}, S_{2i+1}]$, $i \in \{1, .., \frac{m-1}{2}\}$. $IL \equiv \bigcup_{i=1}^{\frac{m-1}{2}} IL_i$ is a minimum feasible insertion list of H.

Proof. As shown for Theorem 9.7. □

10 Edge Connected Triangular Chain Intersection Graphs

In this section we consider an Edge Connected Triangular Chain intersection graph, with $m - 2$ triangular intersection graphs. Each triangular is connected to its neighbors by one different edge, see Figure 20. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and a minimum feasible insertion list.

Lemma 10.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and an edge connected triangular chain intersection graph. $G_{int}(H)$ has $m-2$ triangular induced sub graphs.*

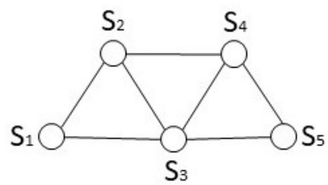

Figure 20: Edge Connected Chain intersection graph

Proof. Proof by induction on t, the number of triangular induced sub graphs in H.

If $t = 1$ then $G_{int}(H)$ contains only one triangular with three clusters, and to check $3 - 2 = 1$ triangle.

Suppose the claim is correct for $t - 1$. We now prove it for t. $G_{int}(H)$ has $t - 1 = m - 3$ triangular induced sub graphs and m clusters. We add one cluster to add one more triangular to $G_{int}(H)$, $t = (m + 1) - 3 = m - 2$. □

Theorem 10.2. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, S_4, S_5\}$ and an edge connected triangular chain intersection graph. If $|X_{1,2,3}| = |X_{2,3,4}| = |X_{3,4,5}|= 1$, then H has a feasible solution tree by paths.*

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 5$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. Let $v_{1,2,3}$ be the only vertex in $X_{1,2,3}$. Let $v_{2,3,4}$ be the only vertex in $X_{1,2,4}$. Let $v_{3,4,5}$ be the only vertex in $X_{1,2,4}$. Figure 21 presents a feasible solution by paths for H.

□

Lemma 10.3. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_i, S_{i+1}, S_{i+2}, S_{i+3}\}$ and an edge connected triangular chain intersection graph. If $H[S_i, S_{i+1}, S_{i+2}]$ is a strongly satisfied triangle on S_i, S_{i+2} and $H[S_{i+1}, S_{i+2}, S_{i+3}]$ is a strongly satisfied triangle on S_{i+1}, S_{i+3}, then H has a feasible solution tree by paths.*

Proof. Consider Figure 22. Let P_i be a path spanning X_i, for $1 \leq i \leq 5$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. If $|X_{i,i+1,i+2}| = 1$, let $v_{i,i+1,i+2}$

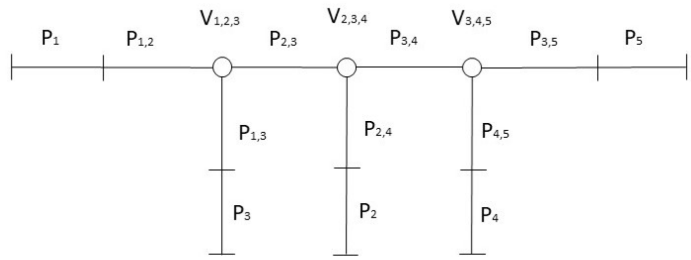

Figure 21: Theorem 10.2 solution tree

be the only vertex in $X_{i,i+1,i+2}$. Otherwise, let $P_{i,i+1,i+2}$ be a path spanning $X_{i,i+1,i+2}$. If $|X_{i+1,i+2,i+3}| = 1$, let $v_{i+1,i+2,i+3}$ be the only vertex in $X_{i+1,i+2,i+3}$. Otherwise, let $P_{i+1,i+2,i+3}$ be a path spanning $X_{i+1,i+2,i+3}$. According to Lemma 5.12, each strongly satisfied triangle has 2 possible solution trees.

1. If $|X_{i,i+1,i+2}| = |X_{i+1,i+2,i+3}| = 1$. Figure 23.1 presents a feasible solution tree by paths for H.

2. If $|X_{i,i+2}| = |X_{i+1,i+3}| = 0$. Figure 23.2 presents a feasible solution tree by paths for H.

3. If $|X_{i+1,i+3}| = 0$ and $|X_{i,i+1,i+2}| = 1$. Figure 23.3 presents a feasible solution tree by paths for H.

4. If $|X_{i,i+2}| = 0$ and $|X_{i+1,i+2,i+3}| = 1$. The construction of the tree is similar to the solution tree shown in figure 23.3.

□

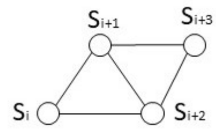

Figure 22: Edge Connected Chain Intersection Graph with four clusters

Lemma 10.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_i, S_{i+1}, S_{i+2}, S_{i+3}\}$ and an edge connected triangular chain intersection graph. If $H[S_i, S_{i+1}, S_{i+2}]$ is a strongly satisfied triangle on S_i, S_{i+2} and $H[S_{i+1}, S_{i+2}, S_{i+3}]$ is a satisfied triangle on S_{i+1}, S_{i+3}, then H has a feasible solution tree by paths.*

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq 5$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. If $|X_{i,i+1,i+2}| = 1$, let $v_{i,i+1,i+2}$ be the only vertex in $X_{i,i+1,i+2}$. Otherwise, let $P_{i,i+1,i+2}$ be a path spanning $X_{i,i+1,i+2}$. If $|X_{i+1,i+2,i+3}| = 1$, let $v_{i+1,i+2,i+3}$ be the only vertex in $X_{i+1,i+2,i+3}$. Otherwise, let $P_{i+1,i+2,i+3}$ be a path spanning $X_{i+1,i+2,i+3}$. According to Lemma 5.12 and 5.14, a strongly satisfied triangle has 2 possible solution tree and a satisfied triangle has 3 possible solution tree.

1. If $|X_{i,i+1,i+3}| = |X_{i+1,i+2,i+3}| = 1$. Figure 24.1 presents a feasible solution tree by paths for H.

2. If $|X_{i+1,i+3}| = 0$ and $|X_{i,i+1,i+3}| = 1$. Figure 24.2 presents a feasible solution tree by paths for H.

3. If $|X_{i+2,i+3}| = 0$ and $|X_{i,i+1,i+3}| = 1$. Figure 24.3 presents a feasible solution tree by paths for H.

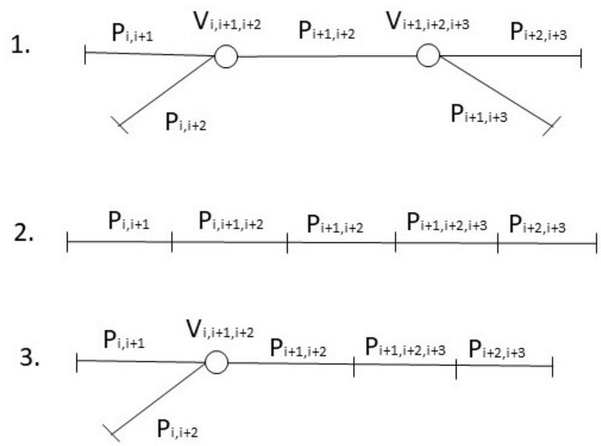

Figure 23: Theorem 10.3 solution tree

4. If $|X_{i,i+2}| = 0$ and $|X_{i+1,i+2,i+3}| = 1$. Figure 24.4 presents a feasible solution tree by paths for H.

5. If $|X_{i,i+2}| = 0$ and $|X_{i+1,i+3}| = 0$. Figure 24.5 presents a feasible solution tree by paths for H.

6. If $|X_{i,i+2}| = 0$ and $|X_{i+2,i+3}| = 0$. Figure 24.6 presents a feasible solution tree by paths for H.

$\square$

Theorem 10.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and edge connected triangular chain intersection graph. H has a feasible solution tree by paths, if the following holds:*

1. $H[S_i, S_{i+1}, S_{i+2}]$ is a strongly satisfied triangle on S_i, S_{i+2}, for $i \in \{2, ..., m-2\}$.

2. $H[S_1, S_2, S_3]$ is a satisfied triangle on S_2, S_3.

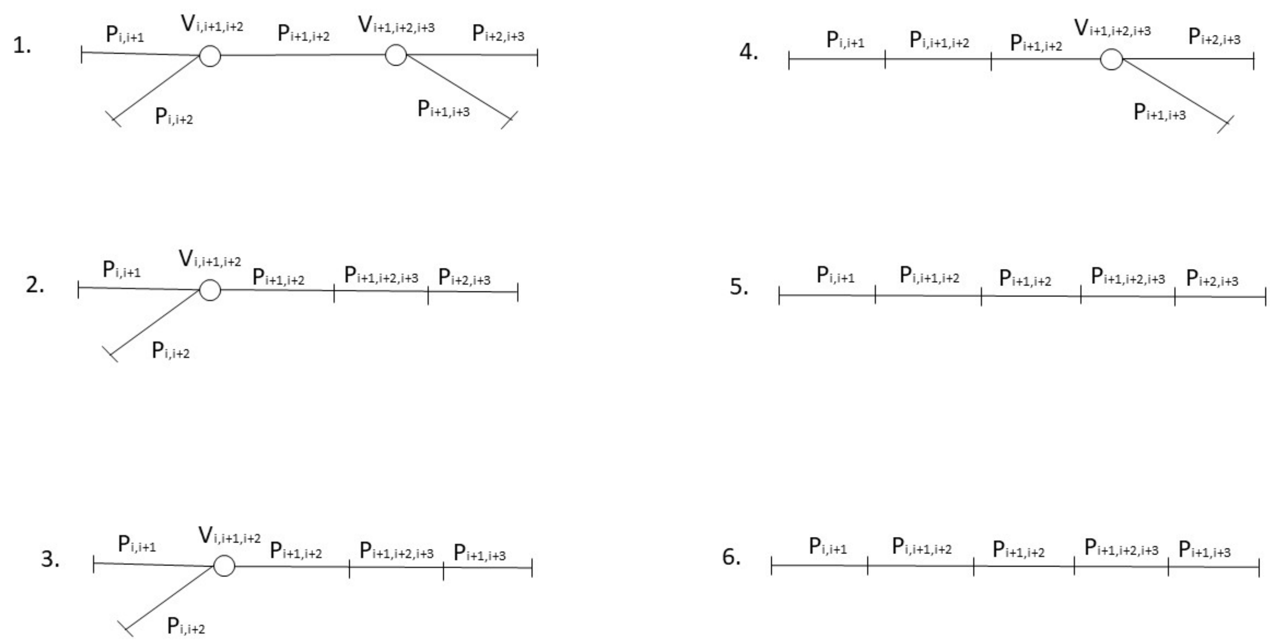

Figure 24: Theorem 10.4 solution tree

Proof. Proof by induction on t, the number of triangular induced sub graphs in $G_{int}(H)$.
If $t = 1$, according to Lemma 10.1, $m = 3$. In this case, $G_{int}(H)$ contains only one triangular intersection graph, which is a satisfied triangle on S_2, S_3, see Figure 25. According to Corollary 5.4, this triangular has a feasible solution tree by paths.

If $t = 2$, according to Lemma 10.1, $m = 4$. In this case, $G_{int}(H)$ is a diamond intersection graph with $\mathcal{S} = \{S_1, ..., S_4\}$, which contains two triangular intersection graphs, see Figure 26. $H[S_1, S_2, S_3]$ is a satisfied triangle on S_2, S_3 and $H[S_2, S_3, S_4]$ is a strongly satisfied triangle on S_2, S_4. According to Corollary 6.11, this diamond has a feasible solution tree by paths.

Suppose the claim is correct for $t - 1$. We now prove it for $t \geq 2$.

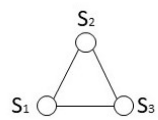

Figure 25: One triangular intersection graph

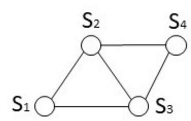

Figure 26: Two triangular intersection graphs

$H[S_1, ..., S_{m-1}]$ has an edge connected chain intersection graph. According to the induction hypothesis, this hypergraph has a feasible solution tree, denote this tree as T. Since $H[S_{m-3}, S_{m-2}, S_{m-1}]$ is a strongly satisfied triangle on S_{m-3}, S_{m-1}, either $|X_{m-3,m-2,m-1}| = 1$ or $|X_{m-3,m-1}| = 0$. According to Lemma 5.12, $T[S_{m-3}, S_{m-2}, S_{m-1}]$ has one of two possible structures, presented in Figure 6, the first corresponds to the case $|X_{m-3,m-2,m-1}| = 1$ and the second to the case $|X_{m-3,m-1}| = 0$.
According to Lemma 5.12, since $H[S_{m-2}, S_{m-1}, S_m]$ is a strongly satisfied triangle on S_{m-2}, S_m, $H[S_{m-2}, S_{m-1}, S_m]$ has a feasible solution tree, denote this tree as T''. According to Lemma 10.3, $H[S_{m-3}, S_{m-2}, S_{m-1}, S_m]$ has a feasible solution tree by paths, denoted as T'. If $|X_{m-3,m-2,m-1}| = 1$ or $|X_{m-3,m-1}| = 0$, then both $T[S_{m-3}, S_{m-2}, S_{m-1}]$ and $T'[S_{m-3}, S_{m-2}, S_{m-1}]$ have the same structure and are therefore identical. In any case, $T[S_{m-3}, S_{m-2}, S_{m-1}] \equiv T'[S_{m-3}, S_{m-2}, S_{m-1}]$ and the two trees T and T'' can be combined into one tree, which is a feasible solution tree by paths of H. $\square$

Theorem 10.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and an edge connected triangular chain intersection graph. H has a feasible* *solution tree by paths, if the following holds:*

1. $H[S_i, S_{i+1}, S_{i+2}]$ *is a strongly satisfied triangle on* S_i, S_{i+2}, *for* $i \in \{2, ..., m-3\}$.

2. $H[S_1, S_2, S_3]$ *is a satisfied triangle on* S_2, S_3.

3. $H[S_{m-2}, S_{m-1}, S_m]$ *is a satisfied triangle on* S_{m-2}, S_{m-1}.

Proof. According to Theorem 10.5, let T be the solution tree for H where $\mathcal{S} = \{S_1, ..., S_{m-1}\}$. Since $H[S_{m-3}, S_{m-2}, S_{m-1}]$ is a strongly satisfied triangle on S_{m-3}, S_{m-1}, either $|X_{m-3,m-2,m-1}| = 1$ or $|X_{m-3,m-1}| = 0$. According to Lemma 5.12, $T[S_{m-3}, S_{m-2}, S_{m-1}]$ has one of two possible structures, presented in Figure 6, the first corresponds to the case $|X_{m-3,m-2,m-1}| = 1$ and the second to the case $|X_{m-3,m-1}| = 0$. According to Lemma 5.14, since $H[S_{m-2}, S_{m-1}, S_m]$ is a satisfied triangle on S_{m-2}, S_{m-1}, $H[S_{m-2}, S_{m-1}, S_m]$ has a feasible solution tree, denote this tree as T''. According to Theorem 10.3, $H[S_{m-3}, S_{m-2}, S_{m-1}, S_m]$ has a feasible solution tree by paths, denoted as T'. If $|X_{m-3,m-2,m-1}| = 1$, then both $T[S_{m-3}, S_{m-2}, S_{m-1}]$ and $T'[S_{m-3}, S_{m-2}, S_{m-1}]$ have the structure presented in Figure 6 and are therefore identical. If $|X_{m-3,m-1}| = 0$, then both $T[S_{m-3}, S_{m-2}, S_{m-1}]$ and $T'[S_{m-3}, S_{m-2}, S_{m-1}]$ have the structure presented in Figure 6 and are therefore identical. In any case, $T[S_{m-3}, S_{m-2}, S_{m-1}] \equiv T'[S_{m-3}, S_{m-2}, S_{m-1}]$ and the two trees T and T'' can be combined into one tree which is a feasible solution tree by paths of H. □

Theorem 10.7. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, ..., S_m\}$ *and an edge connected triangular chain intersection graph. H has no feasible solution tree by paths, if at least one of the following holds:*

1. $H[S_1, S_2, S_3]$ *is not a satisfied triangle on* S_2, S_3.

2. $H[S_{m-2}, S_{m-1}, S_m]$ *is not a satisfied triangle on* S_{m-2}, S_{m-1}.

3. *There is* $i \in \{2, ..., m-3\}$ *such that* $H[S_i, S_{i+1}, S_{i+2}]$ *is not a strongly satisfied triangle on* S_i, S_{i+2}.

Proof. If $H[S_1, S_2, S_3]$ is not a satisfied triangle on S_2, S_3, then according to Corollary 6.11, $H[S_1, S_2, S_3, S_4]$ which has a diamond sub intersection graph does not have a feasible solution tree by paths. Therefore, according to

Lemma 4.2, H does not have a feasible solution tree by paths.

If $H[S_{m-2}, S_{m-1}, S_m]$ is not a satisfied triangle on S_{m-2}, S_{m-1}, then according to Corollary 6.11, $H[S_{m-3}, S_{m-2}, S_{m-1}, S_m]$ which has a diamond sub intersection graph does not have a feasible solution tree by paths. Therefore, according to Lemma 4.2, H does not have a feasible solution tree by paths.

If there is $i \in \{2, ..., m-3\}$ such that $H[S_i, S_{i+1}, S_{i+2}]$ is not a strongly satisfied triangle on S_i, S_{i+2}, in this case $|X_{i,i+1,i+2}| > 1$ and $|X_{i,i+2}| \neq 0$. If in addition $|X_{i,i+1}| \neq 0$ or $|X_{i+1,i+2}| \neq 0$, then $H[S_i, S_{i+1}, S_{i+2}]$ is not a satisfied triangle on S_{i+1}, S_{i+2} or S_i, S_{i+1}, and according to Corollary 6.11, $H[S_i, S_{i+1}, S_{i+2}, S_{i+3}]$ or $H[S_{i-1}, S_i, S_{i+1}, S_{i+2}]$ which have a diamond sub intersection graph do not have a feasible solution tree by paths. Therefore, according to Lemma 4.2, if H has a sub graph that does not have a feasible solution tree by paths, then H does not have a feasible solution tree by paths.

Otherwise, $|X_{i,i+1,i+2}| > 1$, $|X_{i,i+2}| \neq 0$, $|X_{i,i+1}| = 0$ and $|X_{i+1,i+2}| = 0$. Suppose by contradiction that H has a feasible solution tree. Let P_i be a path spanning X_i, for $1 \leq i \leq m$. Let $P_{i,j}$ be a path spanning $X_{i,j}$, for $i \neq j$. Let $P_{i,j,k}$ be a path spanning $X_{i,j,k}$, for $i \neq j \neq k$. According to Lemma 4.7, $P_{i,i+1,i+2}$ has to be connected between $P_{i-1,i,i+1}$ and $P_{i+1,i+2,i+3}$, as shown in Figure 27. According to Lemma 4.5, $P_{i,i+2}$ can not be connected between $P_{i-1,i,i+1}$ and $P_{i,i+1,i+2}$, or between $P_{i,i+1,i+2}$ and $P_{i+1,i+2,i+3}$. According to Lemma 4.6, $P_{i,i+2}$ can not be connected to a vertex connecting $P_{i-1,i,i+1}$ and $P_{i,i+1,i+2}$, or connected to a vertex connecting $P_{i,i+1,i+2}$ and $P_{i+1,i+2,i+3}$. Therefore, $P_{i,i+2}$ can not be connected to $P_{i,i+1,i+2}$ in any way. Therefore, in this case, $P[S_i \bigcap S_{i+2}]$ is not spanned by a connected path, and hence, H has no feasible solution tree by paths.

□

Now we consider removal lists for edge connected triangular chain intersection graph. Note that, if H has a feasible solution tree, every removal list may be empty.

Lemma 10.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and an edge connected triangular chain intersection graph.*
Let $RL^{i,i+1,i+2}$ be a minimum feasible removal list for triangle $H[S_i, S_{i+1}, S_{i+2}]$, such that $H[S_i, S_{i+1}, S_{i+2}] \backslash RL^{i,i+1,i+2}$ is a strongly satisfied triangle on S_i, S_{i+2}, for $i \in \{2, ..., m-3\}$.
Then, $RL^{i,i+1,i+2}$, for $i \in \{2, ..., m-3\}$, are pairwise disjoint.

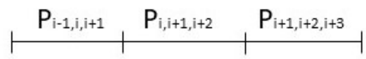

Figure 27: Path: $P_{i-1,i,i+1}$, $P_{i,i+1,i+2}$, $P_{i+1,i+2,i+3}$

Proof. Suppose there is $(v', S') \in RL^{i,i+1,i+2} \bigcap RL^{j,j+1,j+2}$. Obviously, this may happen only if $j = i+1$ or $j = i+2$.
Consider first case $j = i+1$. According to Theorem 5.17, $RL^{i,i+1,i+2} = argmin(|RL_{i,i+2}|, |RL_{i,i+1,i+2}|)$ and $RL^{i+1,i+2,i+3} = argmin(|RL_{i+1,i+3}|, |RL_{i+1,i+2,i+3}|)$.

If $RL^{i,i+1,i+2} = RL_{i,i+2}$ and $RL^{i+1,i+2,i+3} = RL_{i+1,i+2,i+3}$. $RL^{i,i+1,i+2}$ removes vertices $v \in X_{i,i+2}$ from S_i or S_{i+2} and $RL^{i+1,i+2,i+3}$ removes vertices $v \in S_{i+1} \bigcap S_{i+2} \bigcap S_{i+3}$ from either S_{i+1}, S_{i+2} or S_{i+3}. Hence these lists are disjoint.

If $RL^{i,i+1,i+2} = RL_{i,i+2}$ and $RL^{i+1,i+2,i+3} = RL_{i+1,i+3}$. $RL^{i,i+1,i+2}$ removes vertices $v \in X_{i,i+2}$ from S_i or S_{i+2} and $RL^{i+1,i+2,i+3}$ removes vertices from either S_{i+1} or S_{i+3}. Hence these lists are disjoint.

If $RL^{i,i+1,i+2} = RL_{i,i+1,i+2}$ and $RL^{i+1,i+2,i+3} = RL_{i+1,i+3}$. $RL^{i,i+1,i+2}$ removes vertices $v \in S_i \bigcap S_{i+1} \bigcap S_{i+2}$ from either S_i, S_{i+1} or S_{i+2} and $RL^{i+1,i+2,i+3}$ removes vertices $v \in X_{i+1,i+3}$from either S_{i+1} or S_{i+3}. Hence these lists are disjoint.

If $RL^{i,i+1,i+2} = RL_{i,i+1,i+2}$ and $RL^{i+1,i+2,i+3} = RL_{i+1,i+2,i+3}$. $RL^{i,i+1,i+2}$ removes vertices $v \in S_i \bigcap S_{i+1} \bigcap S_{i+2}$ from either S_i, S_{i+1} or S_{i+2} and $RL^{i+1,i+2,i+3}$ removes vertices $v \in S_{i+1} \bigcap S_{i+2} \bigcap S_{i+3}$ from either S_{i+1}, S_{i+2} or S_{i+3}. Hence these lists are disjoint.

Consider case $j = i+2$. According to Theorem 5.17, $RL^{i,i+1,i+2} =$

$argmin(|RL_{i,i+2}|, |RL_{i,i+1,i+2}|)$ and $RL^{i+2,i+3,i+4} = argmin(|RL_{i+2,i+4}|, |RL_{i+2,i+3,i+4}|)$.

If $RL^{i,i+1,i+2} = RL_{i,i+2}$ and $RL^{i+2,i+3,i+4} = RL_{i+2,i+3,i+4}$. $RL^{i,i+1,i+2}$ removes vertices $v \in X_{i,i+2}$ from S_i or S_{i+2} and $RL^{i+2,i+3,i+4}$ removes vertices $v \in S_{i+2} \bigcap S_{i+3} \bigcap S_{i+4}$ from either S_{i+2}, S_{i+3} or S_{i+4}. Hence these lists are disjoint.

If $RL^{i,i+1,i+2} = RL_{i,i+2}$ and $RL^{i+2,i+3,i+4} = RL_{i+2,i+4}$. $RL^{i,i+1,i+2}$ removes vertices $v \in X_{i,i+2}$ from S_i or S_{i+2} and $RL^{i+2,i+3,i+4}$ removes vertices $v \in X_{i+2,i+4}$ from either S_{i+2} or S_{i+4}. Hence these lists are disjoint.

If $RL^{i,i+1,i+2} = RL_{i,i+1,i+2}$ and $RL^{i+2,i+3,i+4} = RL_{i+2,i+4}$. $RL^{i,i+1,i+2}$ removes vertices $v \in S_i \bigcap S_{i+1} \bigcap S_{i+2}$ from either S_i, S_{i+1} or S_{i+2} and $RL^{i+2,i+3,i+4}$ removes vertices $v \in X_{i+2,i+4}$ from either S_{i+2} or S_{i+4}. Hence these lists are disjoint.

If $RL^{i,i+1,i+2} = RL_{i,i+1,i+2}$ and $RL^{i+2,i+3,i+4} = RL_{i+2,i+3,i+4}$. $RL^{i,i+1,i+2}$ removes vertices $v \in S_i \bigcap S_{i+1} \bigcap S_{i+2}$ from either S_i, S_{i+1} or S_{i+2} and $RL^{i+2,i+3,i+4}$ removes vertices $v \in S_{i+2} \bigcap S_{i+3} \bigcap S_{i+4}$ from either S_{i+2}, S_{i+3} or S_{i+4}. Hence these lists are disjoint.

□

Lemma 10.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and an edge connected triangular chain intersection graph.*
Let $RL^{1,2,3}$ be a minimum feasible removal list for triangle $H[S_1, S_2, S_3]$, such that $H[S_1, S_2, S_3] \setminus RL^{1,2,3}$ is a satisfied triangle on S_2, S_3.
Let $RL^{i,i+1,i+2}$ be a minimum feasible removal list for triangle $H[S_i, S_{i+1}, S_{i+2}]$, such that $H[S_i, S_{i+1}, S_{i+2}] \setminus RL^{i,i+1,i+2}$ is a strongly satisfied triangle on S_i, S_{i+2}, for $i \in \{2, ..., m-3\}$.
Then $RL^{1,2,3}$ and $RL^{i,i+1,i+2}$, for $i \in \{2, ..., m-3\}$, are pairwise disjoint.

Proof. Let $RL^{1,2,3} = RL^{j,j+1,j+2}$. Suppose there is $(v', S') \in RL^{1,2,3} \bigcap RL^{j,j+1,j+2}$.
Obviously, this may happen only if $j = 2$ or $j = 3$.
Consider first case $j = 2$.
According to Theorems 5.16 and 5.17, $RL^{2,3,4} = argmin(|RL_{2,4}|, |RL_{2,3,4}|)$ and $RL^{1,2,3} = argmin(|RL_{1,3}|, |RL_{2,3}|, |RL_{1,2,3}|)$.

If $RL^{1,2,3} = RL_{1,3}$ and $RL^{2,3,4} = RL_{2,3,4}$. $RL^{1,2,3}$ removes vertices $v \in X_{1,3}$ from S_1 or S_3 and $RL^{2,3,4}$ removes vertices $v \in S_2 \bigcap S_3 \bigcap S_4$ from either S_2, S_3

or S_4. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{1,3}$ and $RL^{2,3,4} = RL_{2,4}$. $RL^{1,2,3}$ removes vertices $v \in X_{1,3}$ from S_1 or S_3 and $RL^{2,3,4}$ removes vertices $v \in X_{2,4}$ from either S_2 or S_4. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{2,3}$ and $RL^{2,3,4} = RL_{2,3,4}$. $RL^{1,2,3}$ removes vertices $v \in X_{2,3}$ from S_2 or S_3 and $RL^{2,3,4}$ removes vertices $v \in S_2 \bigcap S_3 \bigcap S_4$ from either S_2, S_3 or S_4. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{2,3}$ and $RL^{2,3,4} = RL_{2,4}$. $RL^{1,2,3}$ removes vertices $v \in X_{2,3}$ from S_2 or S_3 and $RL^{2,3,4}$ removes vertices $v \in X_{2,4}$ from either S_2 or S_4. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{1,2,3}$ and $RL^{2,3,4} = RL_{2,3,4}$. $RL^{1,2,3}$ removes vertices $v \in S_1 \bigcap S_2 \bigcap S_3$ from either S_1, S_2 or S_3 and $RL^{2,3,4}$ removes vertices $v \in S_2 \bigcap S_3 \bigcap S_4$ from either S_2, S_3 or S_4. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{1,2,3}$ and $RL^{2,3,4} = RL_{2,4}$. $RL^{1,2,3}$ removes vertices $v \in S_1 \bigcap S_2 \bigcap S_3$ from either S_1, S_2 or S_3 and $RL^{2,3,4}$ removes vertices $v \in X_{2,4}$ from either S_2 or S_4. Hence these lists are disjoint.

Consider case $j = 3$.
According to Theorems 5.16 and 5.17, $RL^{3,4,5} = argmin(|RL_{3,5}|, |RL_{3,4,5}|)$ and $RL^{1,2,3} = argmin(|RL_{1,3}|, |RL_{2,3}|, |RL_{1,2,3}|)$.

If $RL^{1,2,3} = RL_{1,3}$ and $RL^{3,4,5} = RL_{3,4,5}$. $RL^{1,2,3}$ removes vertices $v \in X_{1,3}$ from S_1 or S_3 and $RL^{3,4,5}$ removes vertices $v \in S_3 \bigcap S_4 \bigcap S_5$ from either S_3, S_4 or S_5. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{1,3}$ and $RL^{3,4,5} = RL_{3,5}$. $RL^{1,2,3}$ removes vertices $v \in X_{1,3}$ from S_1 or S_3 and $RL^{3,4,5}$ removes vertices $v \in X_{3,5}$ from either S_3 or S_5. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{2,3}$ and $RL^{3,4,5} = RL_{3,4,5}$. $RL^{1,2,3}$ removes vertices $v \in X_{2,3}$ from S_2 or S_3 and $RL^{3,4,5}$ removes vertices $v \in S_3 \bigcap S_4 \bigcap S_5$ from either S_3, S_4 or S_5. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{2,3}$ and $RL^{3,4,5} = RL_{3,5}$. $RL^{1,2,3}$ removes vertices $v \in X_{2,3}$ from S_2 or S_3 and $RL^{3,4,5}$ removes vertices $v \in X_{3,5}$ from either S_3 or S_5. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{1,2,3}$ and $RL^{3,4,5} = RL_{3,4,5}$. $RL^{1,2,3}$ removes vertices $v \in S_1 \bigcap S_2 \bigcap S_3$ from either S_1, S_2 or S_3 and $RL^{3,4,5}$ removes vertices $v \in S_3 \bigcap S_4 \bigcap S_5$ from either S_3, S_4 or S_5.. Hence these lists are disjoint.

If $RL^{1,2,3} = RL_{1,2,3}$ and $RL^{3,4,5} = RL_{3,5}$. $RL^{1,2,3}$ removes vertices $v \in S_1 \bigcap S_2 \bigcap S_3$ from either S_1, S_2 or S_3 and $RL^{3,4,5}$ removes vertices $v \in X_{3,5}$ from either S_3 or S_5. Hence these lists are disjoint.
Similarly, the proof holds for $RL^{m-2,m-1,m}$ and $RL^{i,i+1,i+2}$, for $i \in \{2, ..., m-3\}$. □

Theorem 10.10. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and an edge connected triangular chain intersection graph.*
Let $RL^{1,2,3}$ be a minimum feasible removal list for triangle $H[S_1, S_2, S_3]$, such that $H[S_1, S_2, S_3] \setminus RL^{1,2,3}$ is a satisfied triangle on S_2, S_3.
Let $RL^{m-2,m-1,m}$ be a minimum feasible removal list for triangle $H[S_{m-2}, S_{m-1}, S_m]$, such that $H[S_{m-2}, S_{m-1}, S_m] \setminus RL^{m-2,m-1,m}$ is a satisfied triangle on S_{m-2}, S_{m-1}.
Let $RL^{i,i+1,i+2}$ be a minimum feasible removal list for triangle $H[S_i, S_{i+1}, S_{i+2}]$, such that $H[S_i, S_{i+1}, S_{i+2}] \setminus RL^{i,i+1,i+2}$ is a strongly satisfied triangle on S_i, S_{i+2}, for $i \in \{2, ..., m-3\}$.
Let $RL = RL^{1,2,3} \bigcup RL^{m-2,m-1,m} \bigcup_{i \in \{2,...,m-3\}} RL^{i,i+1,i+2}$.
RL is a feasible removal list of H.

Proof. In $H \setminus RL$, $RL^{1,2,3}$ and $RL^{m-2,m-1,m}$ are satisfied triangles and $RL^{i,i+1,i+2}$ is a strongly satisfied triangle, for $i \in \{2, ..., m-3\}$. According to Theorem 10.6, $H \setminus RL$ has a feasible solution tree by paths, therefore, RL is a feasible removal list for H. □

Theorem 10.11. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and an edge connected triangular chain intersection graph.*
Let $RL^{1,2,3}$ be a minimum feasible removal list for triangle $H[S_1, S_2, S_3]$, such that $H[S_1, S_2, S_3] \setminus RL^{1,2,3}$ is a satisfied triangle on S_2, S_3.
Let $RL^{m-2,m-1,m}$ be a minimum feasible removal list for triangle $H[S_{m-2}, S_{m-1}, S_m]$, such that $H[S_{m-2}, S_{m-1}, S_m] \setminus RL^{m-2,m-1,m}$ is a satisfied triangle on S_{m-2}, S_{m-1}.
Let $RL^{i,i+1,i+2}$ be a minimum feasible removal list for triangle $H[S_i, S_{i+1}, S_{i+2}]$, such that $H[S_i, S_{i+1}, S_{i+2}] \setminus RL^{i,i+1,i+2}$ is a strongly satisfied triangle on S_i, S_{i+2},

for $i \in \{2, ..., m-3\}$.
Let $RL = RL^{1,2,3} \bigcup RL^{m-2,m-1,m} \bigcup_{i\in\{2,...,m-3\}} RL^{i,i+1,i+2}$.
RL is a minimum feasible removal list of H.

Proof. According to Theorem 10.10, $mRL(H) \leq |RL^{1,2,3}| + |RL^{m-2,m-1,m}| + \sum_{i=2}^{m-3} |RL^{i,i+1,i+2}|$.
Assume RL' is a minimum feasible removal list for H.
Let $RL'^{1,2,3} = RL'[S_1, S_2, S_3]$, $RL'^{m-2,m-1,m} = RL'[S_{m-2}, S_{m-1}, S_m]$ and $RL'^{i,i+1,i+2} = RL'[S_{m-2}, S_{m-1}, S_m]$, for $i \in \{2, ..., m-3\}$.
According to Lemma 4.4, $RL'[S_1, S_2, S_3]$ is a feasible removal list for $H[S_1, S_2, S_3]$. According to Lemma 4.4, $RL'[S_{m-2}, S_{m-1}, S_m]$ is a feasible removal list for $H[S_{m-2}, S_{m-1}, S_m]$. According to Lemma 4.4, $RL'[S_{m-2}, S_{m-1}, S_m]$ is a feasible removal list for $H[S_{m-2}, S_{m-1}, S_m]$, for $i \in \{2, ..., m-3\}$. Since, $RL'^{1,2,3}$, $RL'^{m-2,m-1,m}$ and $RL'^{i,i+1,i+2}$ are pairwise disjoint, the same proofs hold as in Lemmas 10.8, 10.9 and 10.10. Therefore, $|RL'| = |RL'^{1,2,3}| + |RL'^{m-2,m-1,m}| + \sum_{i=2}^{m-3} |RL'^{i,i+1,i+2}|$.
Since, RL' is a feasible removal list, $|RL'| = |RL'^{1,2,3}| + |RL'^{m-2,m-1,m}| + \sum_{i=2}^{m-3} |RL'^{i,i+1,i+2}| \geq |RL^{1,2,3}| + |RL^{m-2,m-1,m}| + \sum_{i=2}^{m-3} |RL^{i,i+1,i+2}|$. □

Now we consider insertion lists for edge connected triangular chain intersection graph. Note that, if H has a feasible solution tree, there is no need for an insertion list.

Lemma 10.12. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, ..., S_m\}$ *and an edge connected triangular chain intersection graph..*
Let $IL^{i,i+1,i+2}$ *be a minimum feasible insertion list for triangle* $H[S_i, S_{i+1}, S_{i+2}]$, *such that* $H[S_i, S_{i+1}, S_{i+2}] + IL^{i,i+1,i+2}$ *is a strongly satisfied triangle on* S_i, S_{i+2}, *for* $i \in \{2, ..., m-3\}$.
Then, $IL^{i,i+1,i+2}$, *for* $i \in \{2, ..., m-3\}$, *are pairwise disjoint.*

Proof. Suppose there is $(v', S') \in IL^{i,i+1,i+2} \bigcap IL^{j,j+1,j+2}$. Obviously, this may happen only if $j = i+1$ or $j = i+2$.
Consider first case $j = i+1$. According to Theorem 5.19, $IL^{i,i+1,i+2} = IL_{(i,i+2)+(i+1)})$ and $IL^{i+1,i+2,i+3} = IL_{(i+1,i+3)+(i+2)})$. In this case, $IL^{i,i+1,i+2}$ inserts vertices from $X_{i,i+2}$ to S_{i+1} and $RL^{i+1,i+2,i+3}$ inserts vertices from $X_{i+1,i+3}$ to S_{i+2}. Hence these lists are disjoint.

Consider case $j = i+2$. According to Theorem 5.19, $IL^{i,i+1,i+2} = IL_{(i,i+2)+(i+1)})$ and $IL^{i+2,i+3,i+4} = IL_{(i+2,i+4)+(i+3)})$. In this case, $IL^{i,i+1,i+2}$ inserts vertices from $X_{i,i+2}$ to S_{i+1} and $RL^{i+2,i+3,i+4}$ inserts vertices from

$X_{i+2,i+4}$ to S_{i+3}. Hence these lists are disjoint.

□

Lemma 10.13. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and an edge connected triangular chain intersection graph..*
Let $IL^{i,i+1,i+2}$ be a minimum feasible insertion list for triangle $H[S_i, S_{i+1}, S_{i+2}]$, such that $H[S_i, S_{i+1}, S_{i+2}]+IL^{i,i+1,i+2}$ is a strongly satisfied triangle on S_i, S_{i+2}, for $i \in \{2, ..., m-3\}$.
Let $IL^{1,2,3}$ be a minimum feasible removal list for triangle $H[S_1, S_2, S_3]$, such that $H[S_1, S_2, S_3] + IL^{1,2,3}$ is a satisfied triangle on S_2, S_3.
Then $IL^{1,2,3}$ and $IL^{i+1,i+2,i+3}$, for $i \in \{2, ..., m-3\}$, are pairwise disjoint.

Proof. Suppose there is $(v', S') \in IL^{1,2,3} \bigcap IL^{j,j+1,j+2}$. Obviously this may happen only if $j = 2$ or $j = 3$.
Consider first case $j = 2$.According to Theorems 5.19 and 5.18, $IL^{2,3,4} = IL_{(2,4)+(3)})$ and $IL^{1,2,3} = argmin(|IL_{(1,3)+(2)}|, |IL_{(1,2)+(3)}|)$.

If $IL^{1,2,3} = IL_{(1,3)+(2)}$. In this case, $IL^{1,2,3}$ inserts vertices from $X_{1,3}$ to S_2 and $RL^{2,3,4}$ inserts vertices from $X_{2,4}$ to S_3. Hence these lists are disjoint.

If $IL^{1,2,3} = IL_{(1,2)+(3)}$. In this case, $IL^{1,2,3}$ inserts vertices from $X_{1,2}$ to S_3 and $RL^{2,3,4}$ inserts vertices from $X_{2,4}$ to S_3. Hence these lists are disjoint.

Consider case $j = 3$. According to Theorems 5.19 and 5.18, $IL^{3,4,5} = IL_{(3,5)+(4)})$ and $IL^{1,2,3} = argmin(|IL_{(1,3)+(2)}|, |IL_{(1,2)+(3)}|)$..

If $IL^{1,2,3} = IL_{(1,3)+(2)}$. In this case, $IL^{1,2,3}$ inserts vertices from $X_{1,3}$ to S_2 and $RL^{3,4,5}$ inserts vertices from $X_{3,5}$ to S_4. Hence these lists are disjoint.

If $IL^{1,2,3} = IL_{(1,2)+(3)}$. In this case, $IL^{1,2,3}$ inserts vertices from $X_{1,2}$ to S_3 and $RL^{3,4,5}$ inserts vertices from $X_{3,5}$ to S_4. Hence these lists are disjoint.
Similarly, the proof holds for $IL^{m-2,m-1,m}$ and $IL^{i,i+1,i+2}$, for $i \in \{2, ..., m-3\}$. □

Theorem 10.14. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and an edge connected triangular chain intersection graph.*
Let $IL^{1,2,3}$ be a minimum feasible insertion list for triangle $H[S_1, S_2, S_3]$, such that $H[S_1, S_2, S_3] \bigcup IL^{1,2,3}$ is a satisfied triangle on S_2, S_3.
Let $IL^{m-2,m-1,m}$ be a minimum feasible insertion list for triangle $H[S_{m-2}, S_{m-1}, S_m]$,

such that $H[S_{m-2}, S_{m-1}, S_m] \bigcup IL^{m-2,m-1,m}$ *is a satisfied triangle on* S_{m-2}, S_{m-1}.
Let $IL^{i,i+1,i+2}$ *be a minimum feasible insertion list for triangle* $H[S_i, S_{i+1}, S_{i+1}]$,
such that $H[S_i, S_{i+1}, S_{i+1}] \bigcup IL^{i,i+1,i+2}$ *is a strongly satisfied triangle on* S_i, S_{i+2},
for $i \in \{2, ..., m-3\}$.
Let $IL = IL^{1,2,3} \bigcup IL^{m-2,m-1,m} \bigcup_{i \in \{2,...,m-3\}} IL^{i,i+1,i+2}$.
IL *is a minimum feasible insertion list of* H.

Proof. The proof is similar to Theorems 10.10 and 10.11. □

11 One Chordless Cycle Intersection Graphs

In this section we consider a One Chordless Cycle intersection graph. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and two minimum feasible insertion lists. The first insertion list, inserts a vertex from each intersection to the same cluster. The second insertion list, inserts the same vertex from an intersection to all the clusters that do not include him.

Theorem 11.1. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, ..., S_m\}$, $m \geq 4$, *and a one chordless cycle intersection graph.* H *has no feasible solution tree by paths.*

Proof. Since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1, H has no feasible solution tree by paths. □

Now we consider removal lists for one chordless cycle intersection graph.

Theorem 11.2. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, ..., S_m\}$ *and a one chordless cycle intersection graph.*
Let $RL = argmin(RL_{1,2}, RL_{2,3}, ..., RL_{m-1,m}, RL_{m,1})$. RL *is a feasible removal list of* H *and is the removal list which removes an edge from* $G_{int}(H)$.

Proof. Without loss of generality, suppose, $RL = RL_{i,i+1}$, for some $1 \leq i \leq m-1$. In $H \setminus RL_{i,i+1}$, $|X_{i,i+1}| = 0$ and in the intersection graph the edge (s_i, s_{i+1}) is removed, so the intersection graph of $H \setminus RL_{i,i+1}$ is a path. Let P_i be a path spanning X_i, for $i \in \{1, .., m\}$. Let $P_{i,i+1}$ be a path spanning $X_{i,i+1}$. Figure 28 presents a feasible solution by paths for $H \setminus RL_{i,i+1}$.
A similar proof applies for case $RL = RL_{m,1}$. □

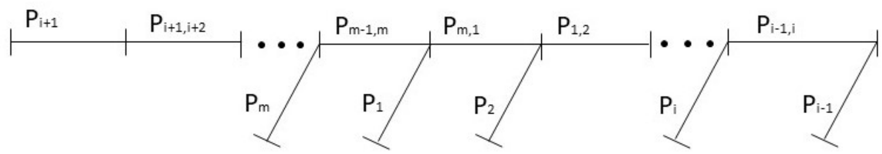

Figure 28: Theorem 11.2 solution tree

Theorem 11.3. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
Let $RL = argmin(RL_{1,2}, RL_{2,3}, ..., RL_{m-1,m}, RL_{m,1})$. RL is a minimum feasible removal list of H.

Proof. Suppose by contradiction, that RL is not a minimum removal list. Let L be a minimum removal list of H. By Theorem 11.2, RL represents the minimum removal list such that, one of the edges of the intersection graph is removed. Since $|L| < |RL|$, no edge was removed from the intersection graph and no edge was added to the intersection graph. Therefore, $H \setminus L$ intersection graph is still a one chordless cycle intersection graph. According to Theorem 11.1, $H \setminus L$ does not have a feasible solution tree by paths. Contradicting the assumption that L is a feasible removal list. □

Theorem 11.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
Let $RL = argmin(RL_{1,2}, RL_{2,3}, ..., RL_{m-1,m}, RL_{m,1})$. RL is the only minimum feasible removal list of H.

Proof. According to Theorem 11.3, RL is a minimum feasible removal list of H. Let L be the minimum removal list of H. All minimum removal lists of H have to remove vertices so that one of $G_{int}(H)$ edges will be removed and $G_{int}(H)$ will not be a one chordless cycle intersection graph. Otherwise, according to Theorem 11.1, $H \setminus L$ does not have a feasible solution tree by paths. RL represents the minimum removal list such that, one of the edges of the intersection graph is removed. Therefore, RL is the only minimum feasible removal list of H. □

Now we consider two minimum insertion lists for one chordless cycle intersection graph.

Definition 11.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
Let $S_j \in \mathcal{S}$ and denote $\mathbf{IL_j} = \{(v_{i,i+1}, S_j)|\ v_{i,i+1} \in X_{i,i+1},$ for $i \in \{1, ...j-2, j+1, ..., m-1\}\}$. Note that, $|IL_j| = m - 2$.

Theorem 11.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
IL_j is a feasible insertion list of H.

Proof. Let P_i be a path spanning X_i, for $1 \leq i \leq m$. Let $P_{i,i+1}$ be a path spanning $X_{i,i+1}$. Let $v_{i,i+1}$ be the vertex chosen from $S_i \bigcap S_{i+1}$. Vertices $v_{i,i+1}$, for $i \in \{1, ...j-2, j+1, ..., m-1\}$, are connected by a path $v_{j+1,j+2}, v_{j+2,j+3}, ..., v_{m-1,m}, ..., v_{j-2,j-1}$, denote this path by P'. Every $P_{i,i+1}$ is connected to the corresponding vertex $v_{i,i+1}$, for $i \in \{1, ..., m-1\}\}$. S_j is spanned by $P_{j,j+1}, P'$ and $P_{j-1,j}$. Figure 29 presents a feasible solution by paths for $H + IL_j$.

□

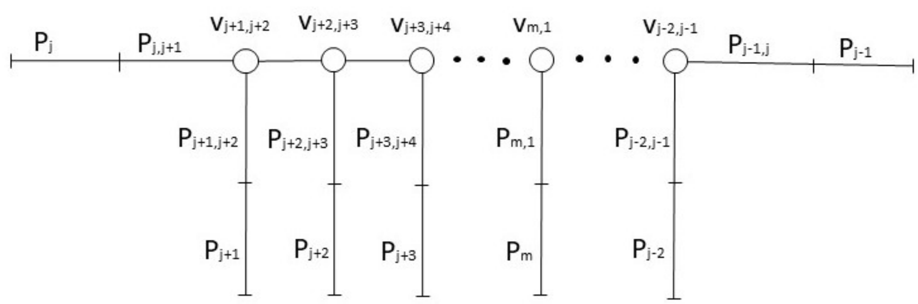

Figure 29: Theorem 11.6 solution tree

Theorem 11.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
IL_j is a minimum feasible insertion list of H.

Proof. According to Theorem 11.6, IL_j is a feasible insertion list of H. According to the definition of IL_j, $|IL_j| = m - 2$. According to Theorem 4.8, and since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1,in every insertion list there are at least $m - 2$ insertions. IL_j is a minimum feasible insertion list of H. □

Definition 11.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
Choose an intersection from $\{X_{1,2}, X_{2,3}, ..., X_{m-1,m}, X_{m,1}\}$, denote by $X_{j,j+1}$.
Choose a vertex $v \in X_{j,j+1}$.
Denote $\mathbf{IL_v} = \{(v, S_1), (v, S_2), (v, S_3), ..., (v, S_{j-1}), (v, S_{j+2}), ..., (v, S_m)\}$.
Note that, $|IL_v| = m - 2$.

Theorem 11.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
IL_v is a feasible insertion list of H, where in $H + IL_v$, $v \in S_i, \forall i$.

Proof. In $H + IL_v$, $v \in S_i$ for every $S_i \in \mathcal{S}$. Let P_i be a path spanning X_i, for $1 \le i \le m$. Let $P_{i,i+1}$ be a path spanning $X_{i,i+1}$. Let v be the chosen vertex. Every $P_{i,i+1}$ is connected in one end point to v and the other end point to P_i, see Figure 30. Figure 30 presents a feasible solution by paths for $H + IL_v$.
□

Theorem 11.10. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a one chordless cycle intersection graph.*
IL_v is a minimum feasible insertion list of H, where in $H + IL_v$, $v \in S_i \forall i$.

Proof. According to Theorem 11.9, IL_v is a feasible insertion list of H. According to the definition of IL_v, $|IL_v| = m - 2$. According to Theorem 4.8, and since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1, in every insertion list there are at least $m - 2$ insertions. IL_v is a minimum feasible insertion list of H. □

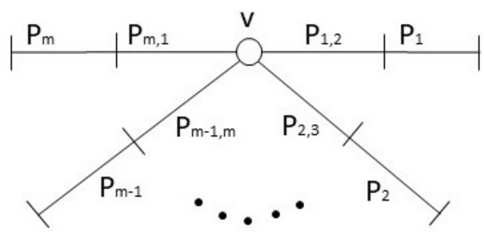

Figure 30: Theorem 11.9 solution tree

12 Two Chordless Cycles With A Separating Edge Intersection Graphs

In this section we consider a Two Chordless Cycle With a Separating Edge intersection graph, see Figure 31. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and two minimum feasible insertion lists. The first insertion list, inserts the same vertex from an intersection to all the clusters that do not include him. The second insertion list, inserts a vertex from each intersection to the same cluster.

Definition 12.1. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph and a two chordless cycles with a separating edge* (s_1, s_2) *intersection graph. The removal of nodes* $\{s_1, s_2\}$ *and edge* (s_1, s_2) *creates two connected components, corresponding to the clusters collections* $\mathcal{S}_a, \mathcal{S}_b$. *Let* $\mathcal{S}_\mathbf{a} = \{\mathbf{R^a_3}, ..., \mathbf{R^a_{m_a}}\}$ *and* $\mathcal{S}_\mathbf{b} = \{\mathbf{R^b_3}, ..., \mathbf{R^b_{m_b}}\}$. *Let* m_a *and* m_b *be the number of clusters in* $\mathcal{S}_a$ *and* $\mathcal{S}_b$*, respectively.*

Theorem 12.2. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph and a two chordless cycles with a separating edge* (s_1, s_2) *intersection graph. If* $max\{m_a, m_b\} \geq 2$*,* H *has no feasible solution tree by paths.*

Proof. If $max\{m_a, m_b\} \geq 2$, at least one of the cycles is a chordless cycle with at least four nodes. Since $CSTP$ is a special case of $CSTT$, then according to Theorem 2.1, H has no feasible solution tree by paths. □

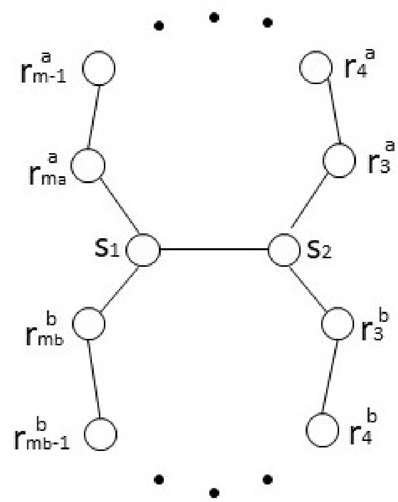

Figure 31: Two Chordless Cycle With A Separating Edge intersection graph

Now we consider removal lists for two chordless cycles with a separating edge (s_1, s_2) intersection graph.

Definition 12.3. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $\mathbf{RL^a_{i,i+1}} = \{(v, R^a_i)|v \in R^a_i \bigcap R^a_{i+1},\ for\ i \in \{3, .., m-1\}\}$ and let $\mathbf{RL^b_{i,i+1}} = \{(v, R^b_i)|v \in R^b_i \bigcap R^b_{i+1},\ for\ i \in \{3, .., m-1\}\}$.

Let $\mathbf{RL^a_{2,3}} = \{(v, S_2)|v \in S_2 \bigcap R^a_3\}$ and let $\mathbf{RL^b_{2,3}} = \{(v, S_2)|v \in S_2 \bigcap R^b_3\}$.

Let $\mathbf{RL^a_{m_a,1}} = \{(v, R^a_{m_a})|v \in R^a_{m_a} \bigcap S_1\}$ and let $\mathbf{RL^b_{m_b,1}} = \{(v, R^b_{m_b})|v \in R^b_{m_b} \bigcap S_1\}$.

Theorem 12.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $RL^{a,b} = argmin(RL^a_{2,3}, ..., RL^a_{m_a-1,m_a}, RL^a_{m_a,1}) \bigcup argmin(RL^b_{2,3}, ..., RL^b_{m_b-1,m_b}, RL^b_{m_b,1})$. $RL^{a,b}$ is a feasible removal list of H.

Proof. $RL^{a,b}$ removes an edge such that both end nodes correspond to clusters from $\mathcal{S}_a$, and an edge such that both end nodes correspond to clusters from $\mathcal{S}_b$, see Figure 32.1 In this case, $G_{int}(H \setminus RL^{a,b})$ is a tree. According to Lemma 4.2, it has a feasible solution tree by paths. □

Theorem 12.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $RL^a = RL_{1,2} \bigcup argmin(RL^a_{2,3}, ..., RL^a_{m_a-1,m_a}, RL^a_{m_a,1})$.
RL^a is a feasible removal list of H.

Proof. RL^a removes the separating edge (s_1, s_2) and an edge such that both end nodes correspond to clusters from $\mathcal{S}_a$, see Figure 32.2 In this case, $G_{int}(H \setminus RL^a)$ is a path. Since a path is a special case of a tree and according to Lemma 4.2, it has a feasible solution tree by paths. □

Theorem 12.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $RL^b = RL_{1,2} \bigcup argmin(RL^b_{2,3}, ..., RL^b_{m_b-1,m_b}, RL^b_{m_b,1})$.
RL^b is a feasible removal list of H.

Proof. RL^b removes the separating edge (s_1, s_2) and an edge such that both end nodes correspond to clusters from $\mathcal{S}_b$, see Figure 32.3 In this case, $G_{int}(H \setminus RL^b)$ is a path. Since a path is a special case of a tree and according to Lemma 4.2, it has a feasible solution tree by paths. □

Theorem 12.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $RL = argmin(RL^{a,b}, RL^a, RL^b)$. RL is a minimum feasible removal list of H.

Proof. According to Theorem 2.1, if $G_{int}(H)$ contains a chordless cycle, H has no feasible solution tree. Therefore, every removal list has to remove at least two edges from the intersection graph, one from each cycle.

1. $RL^{a,b}$ chooses the minimal list, such that the list removes an edge with both end nodes that correspond to clusters from $\mathcal{S}_a$ and an edge with both end nodes that correspond to clusters from $\mathcal{S}_b$, see Figure 32.1.

2. RL^a chooses the minimal list, such that the list removes the separating edge (s_1, s_2) and an edge with both end nodes that correspond to clusters from $\mathcal{S}_a$, see Figure 32.2.

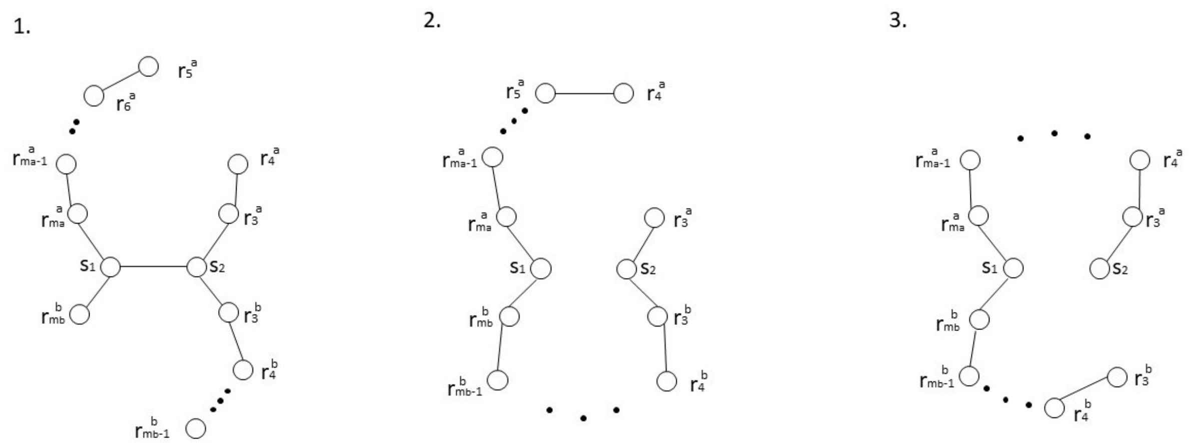

Figure 32: Possible removals

3. RL^b chooses the minimal list, such that the list removes the separating edge (s_1, s_2) and an edge with both end nodes that correspond to clusters from $\mathcal{S}_b$, see Figure 32.3.

RL is a minimum possible option from the three removal lists, so that $H \setminus RL$ has a feasible solution tree by paths. □

Now we consider insertion lists for two chordless cycles with a separating edge (s_1, s_2) intersection graph.

Definition 12.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $\mathbf{X^a_{i,i+1}} = \{(\mathbf{R^a_i} \bigcap \mathbf{R^a_{i+1}})\}$, $X^a_{i,i+1}$ contains the vertices of the intersection of R^a_i and R^a_{i+1}, for $i \in \{3, ..., m-1\}$. Let $\mathbf{X^b_{i,i+1}} = \{(\mathbf{R^b_i} \bigcap \mathbf{R^b_{i+1}})\}$, $X^b_{i,i+1}$ contains the vertices of the intersection of R^b_i and R^b_{i+1}, for $i \in \{3, ..., m-1\}$.

Let $\mathbf{X^a_{2,3}} = \{(\mathbf{S_2} \bigcap \mathbf{R^a_3})\}$, $X^a_{2,3}$ contains the vertices of the intersection of S_2 and R^a_3. Let $\mathbf{X^b_{2,3}} = \{(\mathbf{S_2} \bigcap \mathbf{R^b_3})\}$, $X^b_{2,3}$ contains the vertices of the intersection of S_2 and R^b_3.

Let $\mathbf{X^a_{m,1}} = \{(\mathbf{S_1} \bigcap \mathbf{R^a_{m_a}})\}$, $X^a_{m_a,1}$ contains the vertices of the intersection of S_1 and $R^a_{m_a}$. Let $\mathbf{X^b_{m,1}} = \{(\mathbf{S_1} \bigcap \mathbf{R^b_{m_b}})\}$, $X^b_{m_b,1}$ contains the vertices of the intersection of S_1 and $R^b_{m_b}$.

Definition 12.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Choose arbitrarily an intersection $\{X_{1,2}, X^a_{2,3}, ..., X^a_{m_a-1,m_a}, X^a_{m_a,1}\}$, denote by $X^a_{j,j+1}$. Choose arbitrarily a vertex $v_a \in X^a_{j,j+1}$.
Let $\mathbf{IL_{v_a}} = \{(v_a, S_1), (v_a, S_2), (v_a, R^a_3), ..., (v_a, R^a_{j-1}), (v_a, R^a_{j+2}), ..., (v_a, R^a_{m_a})\}$.

Choose arbitrarily an intersection $\{X_{1,2}, X^b_{2,3}, ..., X^b_{m_b-1,m_b}, X^b_{m_b,1}\}$, denote by $X^b_{j,j+1}$. Choose arbitrarily a vertex $v_b \in X^b_{j,j+1}$.
Let $\mathbf{IL_{v_b}} = \{(v_b, S_1), (v_b, S_2), (v_b, R^b_3), ..., (v_b, R^b_{j-1}), (v_b, R^b_{j+2}), ..., (v_b, R^b_{m_b})\}$.

Definition 12.10. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $\mathbf{IL_{v_a,v_b}} = IL_{v_a} \bigcup IL_{v_b}$.

Theorem 12.11. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph. IL_{v_a,v_b} is a feasible insertion list of H, where in $H + IL_{v_a,v_b}$, $v_a \in R^a_i \forall i$, $v_b \in R^b_i \forall i$ and $v_a, v_b \in S_1 \bigcap S_2$.*

Proof. Let P^a_i be a path spanning X^a_i, for $R^a_i \in \mathcal{S}_a$. Let P^b_i be a path spanning X^b_i, for $R^b_i \in \mathcal{S}_b$. Let $P^a_{i,i+1}$ be a path spanning $X^a_{i,i+1} \forall i$. Let $P^b_{i,i+1}$ be a path spanning $X^b_{i,i+1} \forall i$. Let $P_{1,2}$ be a path spanning $X_{1,2}$. Let v_a, v_b be the chosen vertices from $v_a \in X^a_{j,j+1}$ and $v_b \in X^b_{j,j+1}$. $P_{1,2}$ is connected between v_a and v_b. $P^a_{i,i+1}$ is connected between v_a and P^a_i, $\forall i$. $P^b_{i,i+1}$ is connected between v_b and P^b_i, $\forall i$. Figure 33 presents a feasible solution by paths for $H + IL_{v_a,v_b}$.

□

Theorem 12.12. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph. Let $IL_{v_a,v_b} = IL_{v_a} \bigcup IL_{v_b}$. IL_{v_a,v_b} is a minimum feasible insertion list of H, where in $H + IL_{v_a,v_b}$, $v_a \in R^a_i \forall i$, $v_b \in R^b_i \forall i$ and $v_a, v_b \in S_1 \bigcap S_2$.*

Proof. According to Theorem 12.11, IL_{v_a,v_b} is a feasible insertion list of H.

IL_{v_a} inserts vertex v_a to $m_a - 2$ clusters, according to Theorem 4.8, and since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1, in every insertion list there are at least $m - 2$ insertions, therefore, IL_{v_a} is a minimum feasible insertion list for $H[\mathcal{S}_a]$ which is a one chordless cycle. IL_{v_b} inserts vertex v_b to $m_b - 2$ clusters, according to Theorem 4.8, and since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1, in every

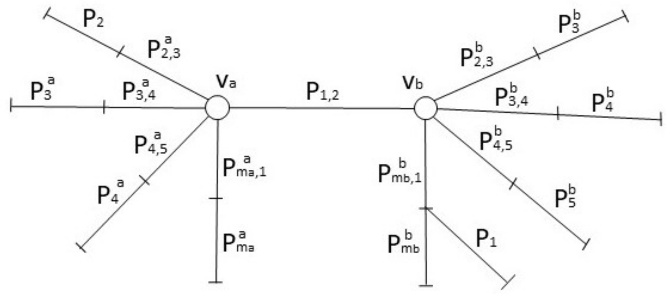

Figure 33: Theorem 12.11 solution tree

insertion list there are at least $m-2$ insertions, therefore, IL_{v_b} is a minimum feasible insertion list for $H[\mathcal{S}_b]$ which is a one chordless cycle. According to [5], $mRL(H) = mRL(H[S_1, S_2, R_3^a, ..., R_{m_a}^a]) + mRL(H[S_1, S_2, R_3^b, ..., R_{m_b}^b]) = |IL_{v_a}| + |IL_{v_b}|$, therefore, IL_{v_a,v_b} is a minimum feasible insertion list of H. $\square$

Definition 12.13. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Choose arbitrarily a vertex $v_{i,i+1}^a \in X_{i,i+1}^a$, for $i \in \{2, ..., m_a - 1\}$.
Let $\mathbf{IL_1^a} = \{(v_{i,i+1}^a, S_1)|$ where $v_{i,i+1}^a \in X_{i,i+1}^a$, for $i \in \{2, ..., m_a - 1\}\}$.
Choose arbitrarily a vertex $v_{i,i+1}^b \in X_{i,i+1}^b$, for $i \in \{2, ..., m_b - 1\}$.
Let $\mathbf{IL_1^b} = \{(v_{i,i+1}^b, S_1)|$ where $v_{i,i+1}^b \in X_{i,i+1}^b$, for $i \in \{2, ..., m_b - 1\}\}$.

Definition 12.14. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph.*
Let $\mathbf{IL_1} = IL_1^a \bigcup IL_1^b$

Theorem 12.15. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph. IL_1 is a feasible insertion list of H.*

Proof. Let P_i^a be a path spanning of X_i^a, for $R_i^a \in \mathcal{S}_a$. Let P_i^b be a path spanning of X_i^b, for $R_i^b \in \mathcal{S}_b$. Let $P_{i,i+1}^a$ be a path spanning of $X_{i,i+1}^a$. Let $P_{i,i+1}^b$ be a path spanning of $X_{i,i+1}^b$. Let $P_{1,2}$ be a path spanning of $X_{1,2}$. Let $v_{1,i,i+1}^a$ be a vertex chosen from $X_{1,i,i+1}^a$, for $i \in \{2, ..., m_a - 1\}$. Let $v_{1,i,i+1}^b$ be a vertex chosen from $X^b_{1,i,i+1}$, for $i \in \{2,...,m_b - 1\}$. All $v^a_{1,i,i+1}$ are connected by a path $v^a_{1,2,3}, v^a_{1,3,4}, ..., v^a_{1,m_a-1,m_a}$ and every intersection spanned by $P^a_{i,i+1}$ is connected to the corresponding vertex $v^a_{1,i,i+1}$. All $v^b_{1,i,i+1}$ are connected by a path $v^b_{1,2,3}, v^b_{1,3,4}, ..., v^b_{1,m_b-1,m_b}$ and every intersection spanned by $P^b_{i,i+1}$ is connected to the corresponding vertex $v^b_{1,i,i+1}$. $P_{1,2}$ is connected between $v^a_{1,2,3}$ and $v^b_{1,2,3}$. Figure 34 presents a feasible solution by paths for $H + IL_1$. □

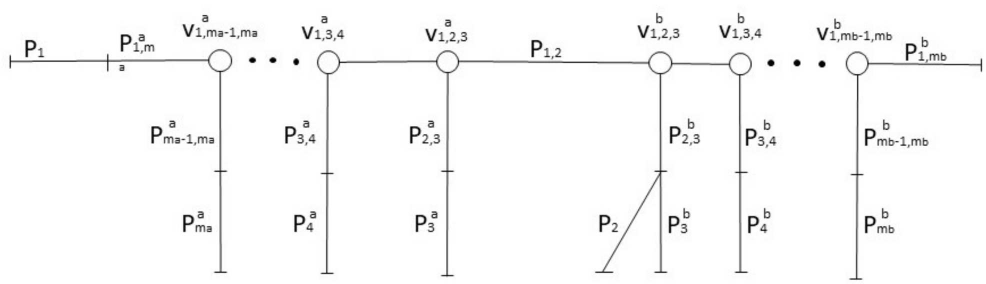

Figure 34: Theorem 12.15 solution tree

Theorem 12.16. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating edge (s_1, s_2) intersection graph. $IL_1 = IL^a_1 \bigcup IL^b_1$ is a minimum feasible insertion list of H.*

Proof. According to Theorem 12.15, IL_1 is a feasible insertion list of H. $|IL^a_1| = m_a - 2$ by definition. According to Theorem 4.8, and since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1, in every insertion list there are at least $m - 2$ insertions. Therefore, IL^a_1 is a minimum feasible insertion list for $H[\mathcal{S}_a]$ which is a one chordless cycle. $|IL^b_1| = m_b - 2$ by definition. According to Theorem 4.8, and since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1, in every insertion list there are at least $m - 2$ insertions. Therefore, IL^b_1 is a minimum feasible insertion list for $H[\mathcal{S}_b]$ which is a one chordless cycle. According to [5], $mRL(H) = mRL(H[S_1, S_2, R^a_3, ..., R^a_{m_a}]) + mRL(H[S_1, S_2, R^b_3, ..., R^b_{m_b}]) = |IL^a_1| + |IL^b_1|$, therefore, IL_1 is a minimum feasible insertion list of H. □

Observation 12.17. *Similarly, Theorems 12.15 and 12.16, hold for* $IL_2 = IL_2^a \bigcup IL_2^b$*, such that* $IL_2^a = \{(v_{2,3}^a, S_2), (v_{3,4}^a, S_2), ..., (v_{m_a-1,m_a}^a, S_2), (v_{m_a,1}^a, S_2)\}$ *and* $IL_2^b = \{(v_{2,3}^b, S_2), (v_{3,4}^b, S_2), ..., (v_{m_b-1,m_b}^b, S_2), (v_{m_b,1}^b, S_2)\}$*.*

13 Two Chordless Cycles With A Separating Path Intersection Graphs

In this section we consider a Two Chordless Cycles With a Separating Path intersection graph, see Figure 35. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and a minimum feasible insertion list.

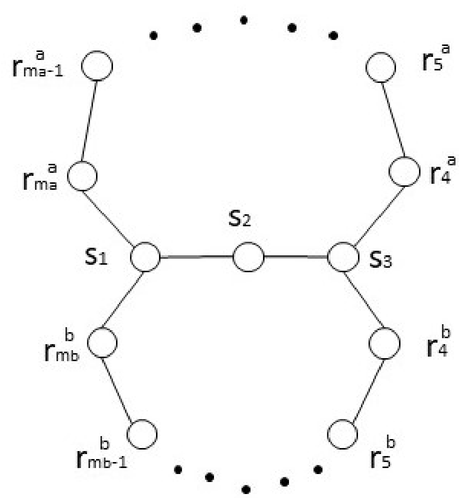

Figure 35: Two Chordless Cycles With a Separating Edge intersection graph

Definition 13.1. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph and two chordless cycles with a separating path* (s_1, s_2, s_3) *intersection graph. The removal of the nodes* $\{s_1, s_2, s_3\}$ *and edges* (s_1, s_2) *and* (s_2, s_3) *creates two connected components corresponding to the clusters collections* $\mathcal{S}_\mathbf{a}, \mathcal{S}_\mathbf{b}$*. Let* $\mathcal{S}_a = \{R_4^a, ..., R_{m_a}^a\}$ *and* $\mathcal{S}_b = \{R_4^b, ..., R_{m_b}^b\}$*, such that* m_a *and* m_b *the number of clusters in* $\mathcal{S}_a$ *and* $\mathcal{S}_b$*, respectively, see Figure 35.*

Theorem 13.2. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph. If $max\{m_a, m_b\} \geq 2$, H has no feasible solution tree by paths.*

Proof. If $max\{m_a, m_b\} \geq 2$, at least one of the cycles is a chordless cycle with at least four nodes. Since $CSTP$ is a special case of $CSTT$, then according to Theorem 2.1, H has no feasible solution tree by paths. □

Now we consider removal lists for two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.

Definition 13.3. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.*
Let $\mathbf{X^a_{i,i+1}} = \{(\mathbf{R^a_i} \bigcap \mathbf{R^a_{i+1}})\}$, $X^a_{i,i+1}$ contains the vertices of the intersection of R^a_i and R^a_{i+1}, for $i \in \{4, ..., m_a - 1\}$. Let $\mathbf{X^b_{i,i+1}} = \{(\mathbf{R^b_i} \bigcap \mathbf{R^b_{i+1}})\}$, $X^b_{i,i+1}$ contains the vertices of the intersection of R^b_i and R^b_{i+1}, for $i \in \{4, ..., m_b - 1\}$.

Let $\mathbf{X^a_{3,4}} = \{(\mathbf{S_3} \bigcap \mathbf{R^a_4})\}$, $X^a_{3,4}$ contains the vertices of the intersection of S_3 and R^a_4. Let $\mathbf{X^b_{3,4}} = \{(\mathbf{S_3} \bigcap \mathbf{R^b_4})\}$, $X^b_{3,4}$ contains the vertices of the intersection of S_3 and R^b_4.

Let $\mathbf{X^a_{m_a,1}} = \{(\mathbf{S_1} \bigcap \mathbf{R^a_{m_a}})\}$, $X^a_{m_a,1}$ contains the vertices of the intersection of S_1 and $R^a_{m_a}$. Let $\mathbf{X^b_{m_b,1}} = \{(\mathbf{S_1} \bigcap \mathbf{R^b_{m_b}})\}$, $X^b_{m_b,1}$ contains the vertices of the intersection of S_1 and $R^b_{m_b}$.

Definition 13.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.*
Let $\mathbf{RL^a_{i,i+1}} = \{(v, R^a_i) | v \in X^a_{i,i+1}\}$ for $i \in \{4, ..., m_a - 1\}$ and let $\mathbf{RL^b_{i,i+1}} = \{(v, R^b_i) | v \in X^b_{i,i+1}\}$, for $i \in \{4, ..., m_b - 1\}$.

Let $\mathbf{RL^a_{3,4}} = \{(v, S_3) | v \in X^a_{3,4}\}$ and let $\mathbf{RL^b_{3,4}} = \{(v, S_2) | v \in X^b_{3,4}\}$.

Let $\mathbf{RL^a_{m_a,1}} = \{(v, R^a_{m_a}) | v \in X^a_{m,1}\}$ and let $\mathbf{RL^b_{m_b,1}} = \{(v, R^b_{m_b}) | v \in X^b_{m,1}\}$.

Definition 13.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.*
Let $\mathbf{RL^{a,b}} = argmin(RL^a_{3,4}, ..., RL^a_{m_a-1,m_a}, RL^a_{m_a,1})$

$\bigcup argmin(RL^b_{3,4}, ..., RL^b_{m_b-1,m_b}, RL^b_{m_b,1})$.

Theorem 13.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.*
$RL^{a,b}$ is a feasible removal list of H.

Proof. $RL^{a,b}$ removes an edge with end nodes that correspond to clusters from $\mathcal{S}_a$ and an edge with end nodes that correspond to clusters from $\mathcal{S}_b$, see Figure 36. In this case, $G_{int}(H \setminus RL^{a,b})$ is a tree. According to Lemma 4.2, it has a feasible solution tree by paths.

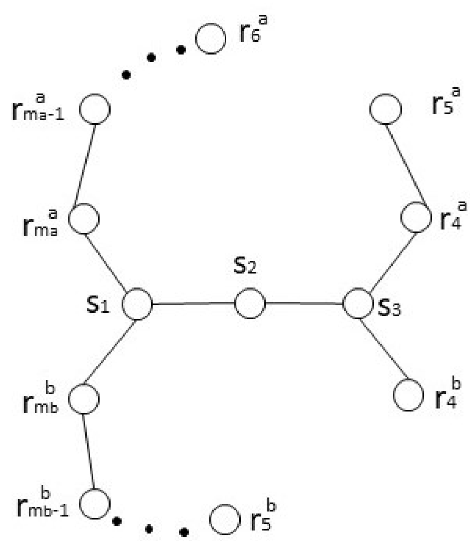

Figure 36: $G_{int}(H \setminus RL^{a,b})$

□

Theorem 13.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.*
Let $RL^a = argmin(RL^a_{3,4}, ..., RL^a_{m_a-1,m_a}, RL^a_{m_a,1}) \bigcup argmin(RL_{1,2}, RL_{2,3})$.
RL^a is a feasible removal list of H.

Proof. RL^a removes an edge with end nodes that correspond to clusters from $\mathcal{S}_a$ and an edge from the separating path $s_1, s_2, s_3)$, see Figure 37. In this case, $G_{int}(H \setminus RL^a)$ is a tree. According to Lemma 4.2, it has a feasible solution tree by paths.

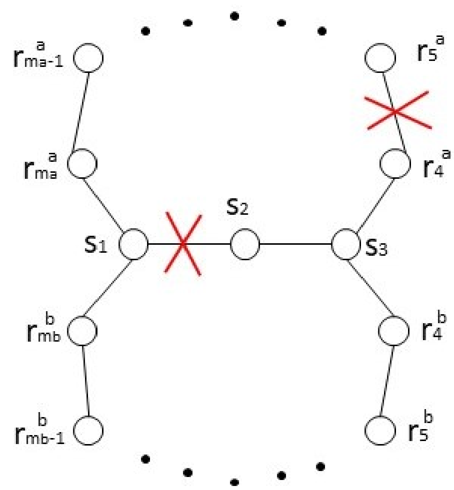

Figure 37: $G_{int}(H \setminus RL^a)$

□

Theorem 13.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.*
Let $RL^b = argmin(RL^b_{3,4}, ..., RL^b_{m_b-1,m_b}, RL^b_{m_b,1}) \bigcup argmin(RL_{1,2}, RL_{2,3})$.
RL^b is a feasible removal list of H.

Proof. RL^b removes an edge with end nodes that correspond to clusters from $\mathcal{S}_b$ and an edge from the separating path (s_1, s_2, s_3), see Figure 38. In this case, $G_{int}(H \setminus RL^b)$ is a tree. According to Lemma 4.2, it has a feasible solution tree by paths.

□

Now we consider insertion lists for two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.

Definition 13.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph and two chordless cycles with a separating path (s_1, s_2, s_3) intersection graph.*

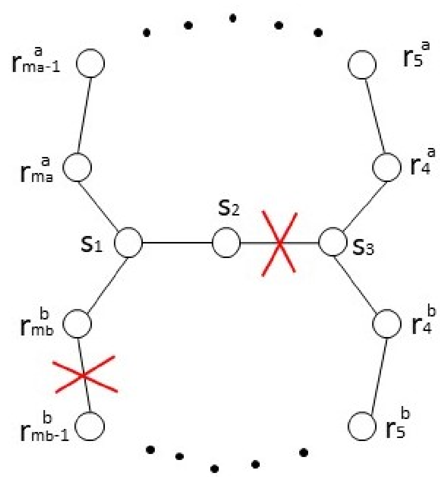

Figure 38: $G_{int}(H \setminus RL^b)$

Choose arbitrarily a vertex $v^a_{i,i+1} \in X^a_{i,i+1}$*, for* $i \in \{3, ..., m_a - 1\}$*. Let* $\mathbf{IL^a_1} = \{(v^a_{3,4}, S_1), ..., (v^a_{m_a-1,m_a}, S_1)\}$*.*

Choose arbitrarily a vertex $v^b_{i,i+1} \in X^b_{i,i+1}$*, for* $i \in \{4, ..., m_b - 1\}$ *and choose arbitrarily a vertex* $v^b_{m_b,1} \in S_1 \bigcap R^b_{m_b}$*. Let* $\mathbf{IL^b_3} = \{(v^b_{4,5}, S_3), ..., (v^b_{m_b,1}, S_3)\}$*.*

Let $\mathbf{IL_{1,2,3}} = (X_{2,3}, S_1) \bigcup (X_{1,2}, S_3)$*.*

Definition 13.10. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph and two chordless cycles with a separating path* (s_1, s_2, s_3) *intersection graph.*
Let $\mathbf{IL^{a,b}_{1,3}} = IL^a_1 \bigcup IL^b_3 \bigcup IL_{1,2,3}$*.*

Theorem 13.11. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph and two chordless cycles with a separating path* (s_1, s_2, s_3) *intersection graph.*
$IL^{a,b}_{1,3}$ *is a feasible insertion list for* H*.*

Proof. Let P^a_i be a path spanning X^a_i, for $R^a_i \in \mathcal{S}_a$. Let P^b_i be a path spanning X^b_i, for $R^b_i \in \mathcal{S}_b$. Let $P^a_{i,i+1}$ be a path spanning $X^a_{i,i+1}$. Let $P^b_{i,i+1}$ be a path spanning $X^b_{i,i+1}$. Let P_1, P_2, P_3 be the paths spanning X_1, X_2, X_3, respectively. Let $P_{1,2,3}$ be the path spanning $X_{1,2,3}$. Let $v^a_{i,i+1}$ be a vertex chosen from $X^a_{1,i,i+1}$. Let $v^b_{i,i+1}$ be a vertex chosen from $X^b_{3,i,i+1}$. All $v^a_{i,i+1}$ are connected by a path $v^a_{3,4}, ..., v^a_{m_a-1,m_a}$ and every intersection spanned by $P^a_{i,i+1}$ is connected to the corresponding vertex $v^a_{i,i+1}$. All $v^b_{i,i+1}$ are connected by a path $v^b_{4,5}, ..., v^b_{m_b-1,m_b}, v^b_{m_b,1}$ and every intersection spanned by $P^b_{i,i+1}$ is connected to the corresponding vertex $v^b_{i,i+1}$. $P_{1,2,3}$ is connected between $v^a_{3,4}$ and $v^b_{m_b,1}$. Figure 39 presents a feasible solution by paths for $H + IL^{a,b}_{1,3}$.

□

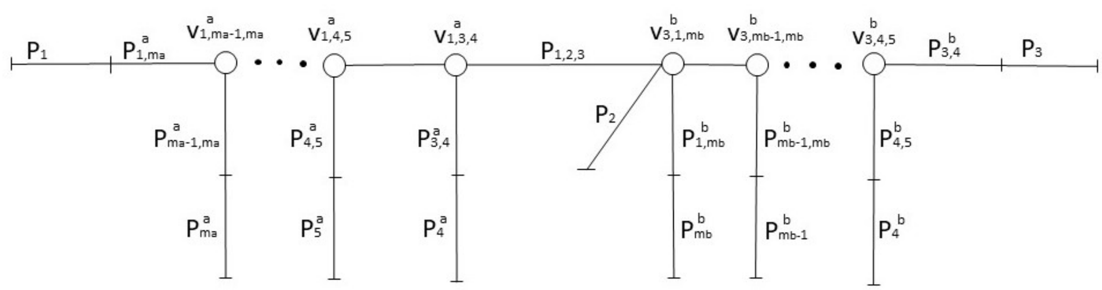

Figure 39: Theorem 13.11 solution tree

14 Triangular Cactus Intersection Graph

In this section we consider a Triangular Cactus Intersection Graph. We describe the conditions for a feasible *CSTP* solution and suggest a minimum feasible removal list and a minimum feasible insertion list.

Definition 14.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph and a triangular cactus intersection graph.*
*$G_{int}(S_i, S_l, S_r)$ is a **triangular leaf** on S_i if $G_{int}(S_i, S_l, S_r)$ is connected to $G_{int}(H \setminus H[S_i, S_l, S_r])$ with only one edge, which touches S_i, see Figure 40.*

Theorem 14.2. *([5]) Consider a hypergraph $H = \langle V, \mathcal{S} \rangle$ with a connected intersection graph $G_{int}(\mathcal{S})$. If node s', whose corresponding cluster is S', is a leaf of $G_{int}(\mathcal{S})$, then H has a feasible solution tree for CSTP problem if and only if $H[\mathcal{S} \setminus S']$ has a feasible solution tree for CSTP problem.*

Theorem 14.3. *([5]) Consider a hypergraph $H = \langle V, \mathcal{S} \rangle$ with T a feasible solution tree for CSTP problem. For any set of vertices $U \subseteq (S_i \backslash (\bigcup_{j \neq i} S_j))$ and $RL_U = \{(U, S_i)\}$, for $S_i \in \mathcal{S}$, hypergraph $H \backslash RL_U$ has a feasible solution tree for CSTP problem.*

Theorem 14.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a triangular cactus intersection graph. If every triangular in $G_{int}(H)$ has a feasible solution tree by paths, then H has a feasible solution tree by paths.*

Proof. Proof by induction on k, the number of nodes in $G_{int}(H)$.

If $k \leq 2$ then $G_{int}(H)$ corresponds to one or two clusters, therefore $G_{int}(H)$ is a tree. According to Lemma 4.2, there exists a feasible solution tree by paths for H.

If $k = 3$ then $G_{int}(H)$ corresponds to three clusters. If $G_{int}(H)$ is a triangle then according to the theorem's assumption, H has a feasible solution tree by paths. Else, $G_{int}(H)$ is a tree and according to Lemma 4.2, has a feasible solution tree by paths for H.

Suppose the claim is correct for $k < m$. We now prove it for $k = m$. If $G_{int}(H)$ has a node s^* which is a leaf, then according to the induction hypothesis $H \setminus S^*$ has a feasible solution tree, and according to Theorem 14.2, H has a feasible solution tree. Otherwise, $G_{int}(H)$ contains a triangular leaf on s_i, denote this triangular as $H[S_i, S_l, S_r]$, see Figure 40. Let $U = S_i \bigcap (S_l \bigcup S_r)$ those vertices are in S_i, but not in $V \setminus (S_i \bigcup S_l \bigcup S_r\})$. According to the induction hypothesis, $H[\mathcal{S} \setminus \{S_l, S_r\}]$ has a feasible solution tree by paths. According to Theorem 14.3, $H[\mathcal{S} \setminus \{S_l, S_r\}] \setminus \{U, S_i\}$ has a feasible solution tree by paths, denote the corresponding tree as T'. Let v be the last vertex in path $T'[S_i \setminus U]$. According to the theorem assumption and Theorem 14.3, $H[S_i, S_l, S_r]$ has a feasible solution tree by paths, and according to Theorem 14.3, $H[U, S_l, S_r]$ also has a feasible solution tree by paths, denoted as T''. According to Corollary 5.4, $H[U, S_l, S_r]$ has four possible solution trees, see Figure 41.

If $|X_{i,r,l}| = 1$ let $v_{i,l,r}$ be the corresponding vertex, else let $P_{U,r,l}$ the path spanning $X_{i,r,l}$. Let $P_{U,r}$ be the path spanning $X_{i,r}$. Let $P_{U,l}$ be the path spanning $X_{i,l}$. Let $P_{l,r}$ be the path spanning $X_{l,r}$. Let P_r be the path spanning X_r. Let P_l be the path spanning X_l. If $|X_{i,r,l}| = 1$, let u be the last vertex in path $P_{U,l}$. Add an edge (v, u) to connect T' and T''. Let T be the new tree (see Figure 41.1). T is a feasible solution tree by paths of H.

Let u be the last vertex in path $P_{U,r}$. Add an edge (v, u) to connect T' and T''. Let T be the new tree (see Figure 41.2). T is a feasible solution tree by

paths of H.
Let u be the last vertex in path $P_{U,l}$. Add an edge (v, u) to connect T' and T''. Let T be the new tree (see Figure 41.3). T is a feasible solution tree by paths of H.
Let u be the last vertex in path $P_{U,r}$. Add an edge (v, u) to connect T' and T''. Let T be the new tree (see Figure 41.4). T is a feasible solution tree by paths of H.

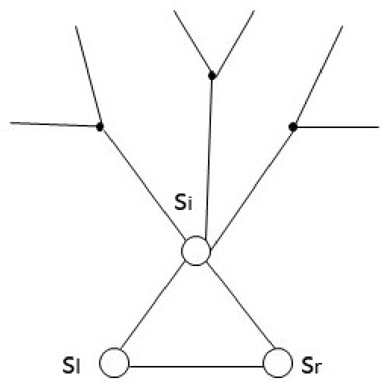

Figure 40: Theorem 14.4 $H[S_i, S_l, S_r]$

□

Theorem 14.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a triangular cactus intersection graph. If there is at least one triangular in $G_{int}(H)$ that does not have a feasible solution tree by paths, then H has no feasible solution tree by paths.*

Proof. According to Lemma 4.2, H does not have a feasible solution tree by paths. □

Now we consider removal list for triangular cactus intersection graph.

Lemma 14.6. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a triangular cactus intersection graph.*
Let n be the number of triangles in $G_{int}(H)$. Let RL_i be a minimum feasible removal list for triangle T_i, $i \in \{1, .., n\}$. RL_i, RL_j are pairwise disjoint, for $i, j \in \{1, .., n\}$, $i \neq j$.

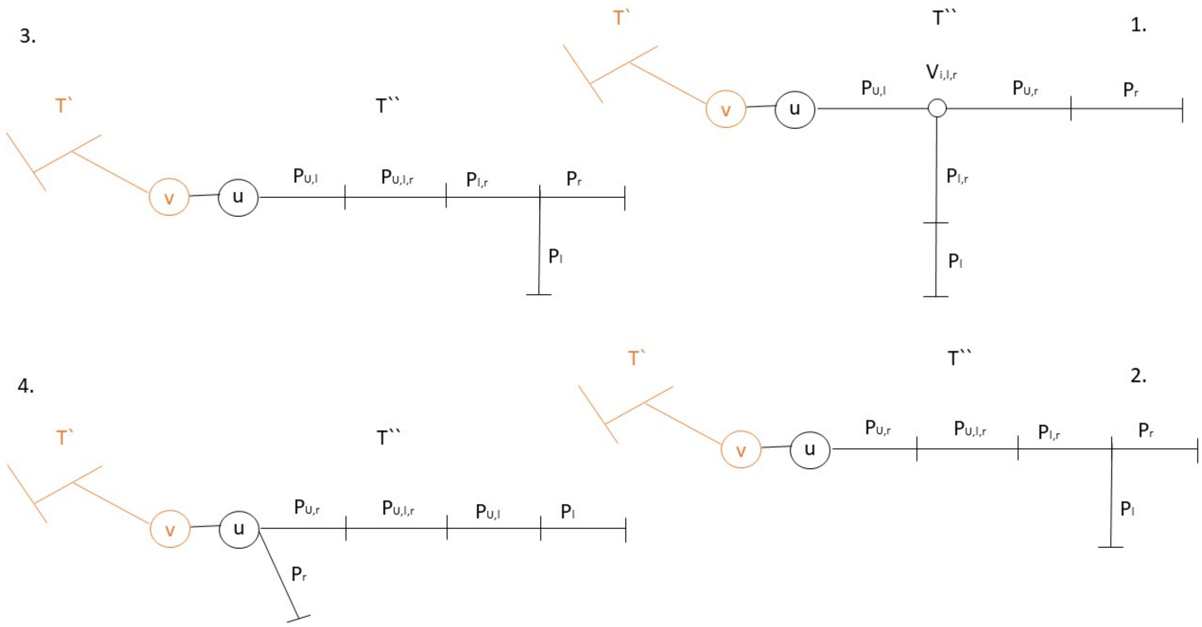

Figure 41: Theorem 14.4 possible solution trees

Proof. Let RL_i be the minimum removal list for $H[S_i, S_{i+1}, S_{i+2}]$ and RL_j be the minimum removal list for $H[S_j, S_{j+1}, S_{j+2}]$. If the intersection graph of $H[S_i, S_{i+1}, S_{i+2}]$ and the intersection graph of $H[S_j, S_{j+1}, S_{j+2}]$ do not have a node in common, then according to Theorem 5.8, RL_i, RL_j are pairwise disjoint. If the intersection graph of $H[S_i, S_{i+1}, S_{i+2}]$ and the intersection graph of $H[S_j, S_{j+1}, S_{j+2}]$ have a node in common, then according to Theorem 5.8, RL_i, RL_j are pairwise disjoint, otherwise, $G_{int}(H)$ has two triangles with two nodes in common. Contradicting the structure of a triangular cactus graph. □

Theorem 14.7. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a triangular cactus intersection graph.*
Let n be the number of triangles in $G_{int}(H)$. Let RL_i be a minimum feasible removal list for triangle Q_i, $i \in \{1, .., n\}$. $RL \equiv \bigcup_{i=1}^{n} RL_i$ is a minimum feasible removal list of H.

Proof. According to Theorem 14.4, if every triangle in the triangular cactus intersection graph has a feasible solution tree by paths, then H has a feasible solution. Since RL_i is a minimum feasible removal list for triangle Q_i, $Q_i \setminus RL_i$, for every $i \in \{1,..,n\}$, has a feasible solution tree. According to Theorem 14.4, $H \setminus \bigcup_{i=1}^{n} RL_i$ has a feasible solution tree. According to Lemma 14.6, $mRL(H) \leq \sum_{i=1}^{n} |RL_i| = \sum_{i=1}^{n} mRL_i$. RL is a feasible removal list of H, therefore, a feasible removal list for Q_i, for every $i \in \{1,..,n\}$, $|RL| = \sum_{i=1}^{n} |RL[Q_i]| \geq \sum_{i=1}^{n} mRL_i$. Hence and according to Theorem 14.4, $RL \equiv \bigcup_{i=1}^{n} RL_i$ is a minimum feasible removal list of H. □

Now we consider insertion lists for triangular cactus intersection graph.

Lemma 14.8. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a triangular cactus intersection graph.*
Let n be the number of triangles in $G_{int}(H)$. Let IL_i be a minimum feasible insertion list for triangle Q_i, $i \in \{1,..,n\}$. IL_i, IL_j are pairwise disjoint, for $i, j \in \{1,..,n\}$, $i \neq j$.

Proof. Let IL_i be a minimum insertion list for $H[S_i, S_{i+1}, S_{i+2}]$ and IL_j be a minimum insertion list for $H[S_j, S_{j+1}, S_{j+2}]$. If the intersection graph of $H[S_i, S_{i+1}, S_{i+2}]$ and the intersection graph of $H[S_j, S_{j+1}, S_{j+2}]$ do not have a node in common, then according to Theorem 5.10, IL_i, IL_j are pairwise disjoint. If the intersection graph of $H[S_i, S_{i+1}, S_{i+2}]$ and the intersection graph of $H[S_j, S_{j+1}, S_{j+2}]$ have a node in common, then according Theorem 5.10, to gain feasibility by using insertions can only be achieved by inserting vertices from $X_{i,i+1}, X_{i,i+2}$ or $X_{i+1,i+2}$ to $X_{i,i+1,i+2}$ and from $X_{j,j+1}, X_{j,j+2}$ or $X_{j+1,j+2}$ to $X_{j,j+1,j+2}$. Therefore, IL_i, IL_j are pairwise disjoint. Otherwise, $G_{int}(H)$ has two triangles with two nodes in common. Contradicting the structure of triangular cactus intersection graph. □

Theorem 14.9. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ and a triangular cactus intersection graph.*
Let n be the number of triangles in $G_{int}(H)$. Let IL_i be a minimum feasible insertion list for triangle Q_i, $i \in \{1,..,n\}$. $IL \equiv \bigcup_{i=1}^{n} IL_i$ is a minimum feasible insertion list of H.

Proof. Let $Q_i = H[S_i, S_{i+1}, S_{i+2}]$. Suppose Q_i does not have a feasible solution, to gain feasibility by using insertions can only be achieved by inserting vertices from $X_{i,i+1}, X_{i,i+2}$ or $X_{i+1,i+2}$ to $X_{i,i+1,i+2}$. According to Lemma 14.8, IL_i, IL_j, for $i, j \in \{1,..,n\}$, $i \neq j$ are pairwise disjoint. Since IL_i is a minimum feasible insertion list for triangle Q_i, $Q_i + IL_i$, for every $i \in \{1,..,n\}$, has a feasible solution tree. According to Theorem 14.4, $H + \bigcup_{i=1}^{n} IL_i$ has a feasible solution tree. According to Lemma 14.8, $mIL(H) \leq \sum_{i=1}^{n} |IL_i| = \sum_{i=1}^{n} mIL_i$. IL is a feasible removal list of H, therefore, a feasible insertion list for Q_i, for every $i \in \{1,..,n\}$, $|IL| = \sum_{i=1}^{n} |IL[Q_i]| \geq \sum_{i=1}^{n} mIL_i$. Hence, $\bigcup_{i=1}^{n} IL_i$ is a minimum feasible insertion list. □

15 Cactus Intersection Graphs

In this section we consider a Cactus Intersection Graph with cycles with length at least 4, see Figure 42. We describe the conditions for a feasible *CSTP* solution and suggest a removal list.

Definition 15.1. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph and a cactus intersection graph.*
$G_{int}(S_i, ..., S_r)$ *is a* ***cycle leaf*** *on* S_i *if* $G_{int}(S_i, ..., S_r)$ *is connected to* $G_{int}(H \setminus H[S_i, ..., S_r])$ *with only one edge, which touches* S_i*, see Figure 42.*

Theorem 15.2. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, ..., S_m\}$*,* $m \geq 4$ *and a cactus intersection graph. If* $G_{int}(H)$ *has at least one cycle with length at least* 4*,* H *has no feasible solution tree by paths.*

Proof. Since $CSTP$ is a special case of $CSTT$ and according to Theorem 2.1, H has no feasible solution tree by paths. □

Theorem 15.3. *Let* $H =< V, \mathcal{S} >$ *be a hypergraph, with* $\mathcal{S} = \{S_1, S_2, S_3, ..., S_m\}$ *and a cactus intersection graph. Let* n *be the number of cycles in* $G_{int}(H)$*. Let* RL_i *be a feasible removal list for cycle* C_i*,* $i \in \{1,..,n\}$*.* $RL \equiv \bigcup_{i=1}^{n} RL_i$ *is a feasible removal list of* H*.*

Proof. Proof by induction on k, the number of nodes in $G_{int}(H)$.

If $k \leq 4$, if $G_{int}(H)$ is a tree, then $RL = \emptyset$ and according to Theorem 4.2, H has a feasible solution tree by paths. If $G_{int}(H)$ is a cycle, then there is only one cycle C_1, such that RL_1 is a minimum feasible removal list for C_1 and according to the theorem assumption $H \setminus RL$ has a feasible solution tree by paths

Suppose the claim is correct for $k < m$. We now prove it for $k = m$. If $G_{int}(H)$ has a node s' which is a leaf. Let $C_1, ..., C_n$ be the cycles in H. Since s' is a leaf, $C_1, ..., C_n$ are also cycles in the intersection graph of $H[\mathcal{S} \setminus S']$, then according to the induction hypothesis $H[\mathcal{S}\backslash S'] \setminus RL$ has a feasible solution tree, and according to Theorem 14.2, $H \setminus RL$ has a feasible solution tree. Otherwise, $G_{int}(H)$ contains a cycle leaf on s_i, denote this cycle as $H[S_i, S_l, ..., S_r, S_i]$ and suppose this is cycle C_n, see Figure 42.

Let $U = S_i \bigcap(\bigcup_{j=l}^{r} S_j)$, these vertices are in S_i, but not in $V(\mathcal{S}\backslash\{S_i \bigcup S_l \bigcup .. \bigcup S_r\})$. According to the induction hypothesis, $H[\mathcal{S} \setminus \{S_l, ..., S_r\}] \setminus \bigcup_{i=1}^{n} RL_i$ has a feasible solution tree by paths. According to Theorem 14.3, $H[\mathcal{S} \setminus \{S_l, S_r\} \setminus (U, S_i)] \setminus \bigcup_{i=1}^{n} RL_i$ has a feasible solution tree by paths, denoted as T'. Let v be the last vertex in the path $T'[S_i \setminus U]$. According to the theorem assumption and Theorem 14.3, $H[S_i, S_l, ..., S_r] \setminus RL_n$ has a feasible solution tree by paths, then $H[U, S_l, ..., S_r] \setminus RL_n$ also has a feasible solution tree by paths, denoted as T''. According to Theorem 11.2, RL_n represents the removal of one of the edges of the cycle corresponding to $H[S_i, S_l, ..., S_r, S_i]$, so that, $H[S_i, S_l, ..., S_r, S_i] \setminus RL_n$ has a solution which is a path. Let u be the last vertex in this path, such that $u \in U$. Add an edge (v, u) to connect T' and T''. Let T be the new tree, see Figure 43. T is a feasible solution tree by paths of $H \setminus RL$. Hence, $H \setminus RL$ has a feasible solution tree by paths.

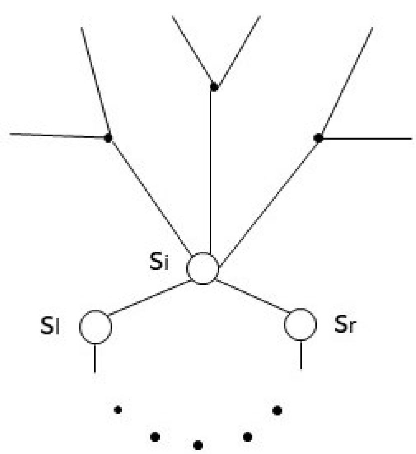

Figure 42: A cycle leaf

□

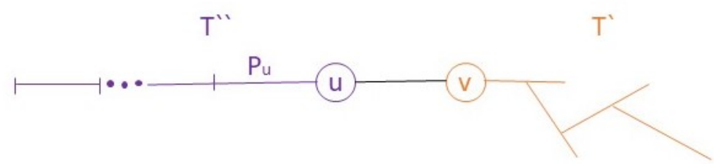

Figure 43: Theorem 15.3 solution tree

16 Algorithm For Solving Triangle Free Graph

In this section we consider a Triangle Free Intersection Graph, a graph which does not contain any triangles. Hence, every cycle in this graph contains at least 4 nodes. We describe the conditions for a feasible $CSTP$ solution and introduce an algorithm for finding a minimum feasible removal list.

Theorem 16.1. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a triangle free intersection graph. If $G_{int}(H)$ has at least one cycle, H has no feasible solution tree by paths.*

Proof. Since $G_{int}(H)$ is a triangle free intersection graph then at least one cycle contains at least 4 nodes. Since $CSTP$ is a special case of $CSTT$ according to Theorem 2.1, H has no feasible solution tree by paths. □

Definition 16.2. *A* ***maximum spanning tree (M_xST)*** *a spanning tree whose weight (the sum of weights of its edges) is maximum.*

Theorem 16.3. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a triangle free intersection graph. Let S_i, S_j, S_k be clusters in $\mathcal{S}$, for every $i, j, k \in \{1, ..., m\}$ $|X_{i,j,k}| = 0$.*

Proof. Suppose by contradiction that, for $i, j, k \in \{1, ..., m\}$, $|X_{i,j,k}| > 0$. In $G_{int}(H)$ nodes s_i, s_j, s_k form a triangle shape, in contradiction to $G_{int}(H)$ being a triangle free intersection graph. Furthermore, every cluster can have at most one intersection with another cluster in H. □

Algorithm 3: TrianglesFreeMinRemovalList

Input : Triangle free intersection graph
Output: Minimum removal list for triangle free intersection graph
$CRL = [\,]$;
Set $w_{i,j} = |X_{i,j}|$ to be the weight of edge (s_i, s_j), for $(s_i, s_j) \in G_{int}(H)$;
Let G_w be $G_{int}(H)$ with weights;
Let T_{max} be a maximum spanning tree of G_w;
Let $E_{rm} = \{(s'_{i_1}, s''_{i_1}), ..., (s'_{i_k}, s''_{i_k})\}$ be the set of edges which are in G_w and not in T_{max};
Let $CRL = \bigcup_{j=1}^{k}(S'_{i_j} \bigcap S''_{i_j}, S'_{i_j})$
return CRL;

Theorem 16.4. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a triangle free intersection graph. Algorithm TrianglesFreeMinRemovalList returns a feasible removal list for H.*

Proof. According to the algorithm, the removal of CRL corresponds to the removal of all the edges in E_{rm} from G_w, thus changing the intersection graph into a tree. Therefore, the intersection graph of $H \setminus CRL$ is a tree, according to Lemma 4.2, it has a feasible solution tree by paths. □

Theorem 16.5. *Let $H =< V, \mathcal{S} >$ be a hypergraph, with $\mathcal{S} = \{S_1, ..., S_m\}$ and a triangle free intersection graph. Algorithm TrianglesFreeMinRemovalList returns a minimum feasible removal list for H.*

Proof. If RL is a feasible removal list, then $G_{int}(H \setminus RL)$ contains no cycles. Otherwise, if $G_{int}(H \setminus RL)$ contains a cycle with at least 4 nodes then according to Theorem 16.1, H does not have a feasible solution tree. According to Theorem 11.4, any feasible removal list removes at least one edge from each cycle, so that $G_{int}(H \setminus RL)$ will be cycles free. In addition, if RL is a minimum feasible removal list, $G_{int}(H \setminus RL)$ is a connected graph. Otherwise, the minimum feasible removal list would have removed one edge less, in contradiction to RL being a minimum feasible removal list. Therefore, if RL is a minimum feasible removal list then $G_{int}(H \setminus RL)$ is a tree, by removing edges from the intersection graph. Finding the set of edges with minimum weight, whose removal from the intersection graph creates a tree, is equivalent to finding a maximum spanning tree. According to Theorem

16.4, CRL is a feasible removal list and represents the removals made to gain a maximum spanning tree. Therefore, CRL is a minimum feasible removal list for H. □

17 Summary and Further Research

Given a hypergraph, the research considers and investigates intersection graph of specific shapes, for each shape we describe the conditions for feasibility regarding a *CSTP* solution. When there is no feasible solution we suggest a minimum feasible removal list and a minimum feasible insertion list. The research starts by looking at intersection graphs with triangular base shapes, such as a triangular, diamond, butterfly, windmill, vertex connected triangular chain and an edge connected triangular chain. The research deals with intersection graphs with special characteristics, where it is easy to show that there is no feasible solution for the given hypergraph. The first intersection graph is a single chordless cycle, followed by an intersection graph with two chordless cycles connected by separating edge or a separating path of size three. A significant part of the research focus on intersection graph which is a triangular cactus tree. We describe the conditions for a feasibility and suggest a minimum removal list or a minimum insertion list. When the intersection graph is a cactus tree, we suggest a minimum removal list. We also provide an algorithm that finds a minimum feasible removal list for a triangular free intersection graph.

We would like to continue our research and investigate more complex structures of intersection graphs, for example a 4-clique. Find conditions for feasibility and suggest a minimum feasible removal list and a minimum feasible insertion list.

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, 1976.

[2] U. Brandes, S. Cornelsen, B. Pampel, and A. Sallaberry. Path-based supports for hypergraphs. *Journal of Discrete Algorithms*, 14:248–261, 2012.

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

[4] C. Flament. Hypergraphes arborés. *Discrete Math.*, 21(3):223–227, 1978.

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

[6] D. S. Johnson and H. O. Pollak. Hypergraph planarity and the complexity of drawing venn diagrams. *Journal of Graph Theory*, 11(3):309–325, 1987.

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

[8] 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.

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

[10] R. Swaminathan and D. K. Wagner. On the consecutive-retrieval problem. *SIAM J. Comput.*, 23(2):398–414, 1994.

[11] A. S. Tanenbaum and D. J. Wetherall. *Computer Networks.* Prentice Hall, 5 edition, 2011.