Seven Edge Forest Designs

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Daniel Mauricio Banegas

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Dr. Bryan Freyberg

May, 2025

Acknowledgements

There are many people that I am grateful to for their contribution to my time in the graduate program at the University of Minnesota Duluth. First and foremost, I am grateful to my family who has supported me extensively throughout the ups and downs of my academic journey. Next I would like to thank my friends on West 7th in Saint Paul, who guided me and taught me everything I need to know in order to succeed in life. Also, I would like to thank Professor Gallian and Professor Froncek who have really influenced how I organize myself mathematically today. Lastly, I would like to thank my advisor Professor Freyberg who has supported and believed in me since I first attended his Discrete Math class in 2019. He has been the most influential presence in my development as a math student and as a mathematician, and I truly wouldn't be either of those if it weren't for him.

Dedication

I dedicate this thesis to my advisor Professor Bryan Freyberg, to my family who has supported me throughout this process, and to Jers, Aaron, Mike B, Joe, Max, Jordi, Ian, TK, Parker, Jehan and Torta from Tuscarora and West 7th in Saint Paul. Thank you for believing in me when I couldn't and helping me realize what is possible when I apply myself.

Abstract

Let G be a subgraph of K_n where $n \in \mathbb{N}$. A G-decomposition of K_n, or G-design of order n, is a finite collection $\{G_1, \ldots, G_k\}$ of pairwise edge-disjoint subgraphs of K_n that are all isomorphic to some graph G and whose union is K_n. We prove that an F-decomposition of K_n exists for every seven edge forest F if and only if $n \equiv 0, 1, 7,$ or 8 (mod 14) and $n \geq 14$. We also share some additional results on edge mappings and galaxy decompositons of complete bipartite graphs along with some `python` programs related to graphs and decompositions.

Contents

List of Tables

List of Figures

Chapter 1

Background

1.1 Fundamentals of Graph Theory

Graph Theory is the study of objects called *vertices* or *nodes* and their relationships which we call *edges*. An edge between vertices u and v is typically denoted uv, (u, v), $\{u, v\}$. A graph G is formally defined as an ordered pair $G = (V, E)$ where V is the set of all vertices in G and E is the set of all edges between vertices in G. These sets are sometimes denoted $V(G)$ and $E(G)$, respectively.

G is called a *simple graph* if (1) there is at most one edge between any two vertices, (2) there are no edges from a vertex to itself and (3) all edges have no directionality to them, meaning $uv = vu$ for any edge $uv \in E(G)$. For the rest of this paper all graphs we work with are finite simple graphs, but note that many of the proceeding definitions are defined in the same or very similar ways for infinite and directed graphs.

Graphs are more intuitive to work with through their visual representations instead of their formal definitions. Let G be a simple graph where $V(G) = \{A, B, C, D, E, a, b, c, d, e\}$ and $E(G) = \{Aa, Bb, Cc, Dd, Ee, AB, BC, CD, DE, EA, ac, ce, eb, bd, da\}$. this graph is known as the *Petersen* graph. It's a bit unwieldy when described formally, yet its visual representation is very easy to understand.

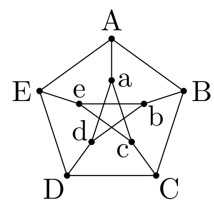

Figure 1.1: The Petersen graph.

We say two graphs G and H are *isomorphic* if there exists a bijection from $V(G)$ to $V(H)$ called an *isomorphism* that induces a bijection from $E(G)$ to $E(H)$ and we denote this relationship via $G \cong H$. In other words, we consider two graphs G and H to be the 'same' if we can relabel and move vertices in some fashion (without adding/removing vertices and edges) in visual representations of G and H to go between the two.

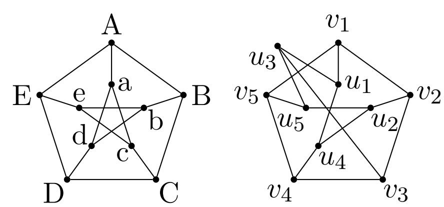

Figure 1.2: (left) $G \cong H$ (right).

Graph theorists generally consider two graphs to be the same if they are isomorphic because an isomorphism preserves all meaningful structural properties of a graph. We will wrap up the fundamentals with a few definitions and some important operations.

Definition 1.1.1 (Subgraph)**.** A subgraph $G \subseteq K$ is a graph whose vertices and edges are subsets of the vertices and edges of K; $G \subseteq K$ if $V(G) \subseteq V(K)$ and $E(G) \subseteq E(K)$.

Definition 1.1.2 (Vertex-induced Subgraph)**.** A *vertex-induced* subgraph $G \subseteq K$ is one whose vertices are some subset S of $V(K)$ and whose edges are all edges between those vertices in K; $V(G) = S \subseteq V(K)$ and $E(G) = \{uv \in E(K) \mid u, v \in S\}$. If G is such a subgraph we say that G is induced by $S = V(G) \subseteq V(K)$.

Definition 1.1.3 (Edge-induced Subgraph)**.** An *edge-induced* subgraph $G \subseteq K$ is one whose edges are some subset of $E(K)$ and whose vertices are all those who appear as an endpoint in that subset of edges; $E(G) \subseteq E(K)$ and $V(G) = \{u \in V(K) \mid uv \in E(G) \text{ for some } v \in V(K)\}$. If G is such a subgraph we say that G is induced by $S = E(G) \subseteq E(K)$

Here is a visual example of these types of graphs: Let K be the Petersen graph from Figure 1.1.

Subgraph: $G \subseteq K$ where $V(G) = \{E, e, b\}$, $E(G) = \{Ee\}$.

Vertex-induced Subgraph: $H \subseteq K$ is induced by $\{a, A, B\} \subseteq V(K)$

Edge-induced Subgraph: $M \subseteq K$ is induced by $\{Dd, DC, Cc\} \subseteq E(K)$

The figure below shows K and it's color-coded subgraphs.

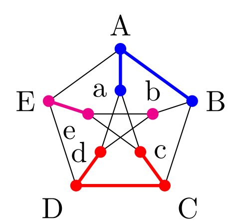

Figure 1.3: K and subgraphs $G, H, M \subseteq K$.

Next, we will talk about some important operations done on graphs.

Definition 1.1.4 (Graph Complement)**.** The complement of a graph G denoted $\overline{G}$ is the graph obtained by removing all edges of G and then adding in all edges not originally present in G. Formally,

$$\overline{G} = (V(G), \{uv \mid u, v \in V(G) \text{ and } uv \notin G\})$$

Here is an example of a graph complement. Let $G = (\{a, b, c, d\}, \{ab, bc, cd, da\})$. Then $\overline{G} = (\{a, b, c, d\}, \{ac, bd\})$. These graphs are depicted in the figure below.

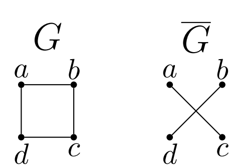

Figure 1.4: (left) G and $\overline{G}$ (right).

Definition 1.1.5 (Graph Union)**.** The union of two graphs G and H is simply the graph resulting from the union of their vertices and the union of their edges and is denoted $G \cup H$; $G \cup H = (V(G) \cup V(H), E(G) \cup E(H))$. If G and H are vertex-disjoint, we may denote their union via $G \sqcup H$ and call it a *disjoint union* of G and H. If they are only edge-disjoint, we may call it an *edge-disjoint union.* We also use $\sqcup$ in general to denote a union between disjoint sets; $A \sqcup B$ to denotes a union between sets A and B where $A \cap B = \emptyset$.

Here is an example of a union and a disjoint union of graphs. Let $G = (\{a, b, c, d\}, \{ab, bc, cd, da\}), H = (\{a, b, c\}, \{ab, bc, ca\}),$ and $K = (\{A, B, C\}, \{AB, BC, CA\})$ Then:

$$G \cup H = (\{a, b, c, d\}, \{ab, bc, cd, da, ca\})$$
$$G \sqcup K = (\{a, b, c, d, A, B, C\}, \{ab, bc, cd, da, AB, BC, CA\})$$

These unions are depicted in the following figure.

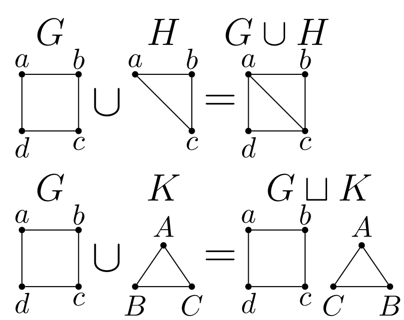

Figure 1.5: (above) $G \cup H$ and (below) $G \sqcup K$.

Next, we define another very important operation that combines two vertex disjoint graphs in a different manner.

Definition 1.1.6 (Join)**.** Let G and H be vertex disjoint graphs. Their *join*, denoted $G \vee H$, is the graph obtained by taking the disjoint union of G and H and adding all possible edges between every vertex in G and every vertex in H. Formally:

$$G \vee H \;=\; (V(G) \cup V(H), E(G) \;\cup\; E(H) \;\cup\; \{\, xy \mid x \in V(G),\, y \in V(H)\}).$$

Here is an example. Let $G = (\{a, b, c\}, \{ab, bc, ca\})$ and $H = (\{A, B, C\}, \{AB, BC, CA\})$, then $G \vee H = (\{a, b, c, A, B, C\}, E(G) \sqcup E(H) \sqcup \{aA, aB, aC, bA, bB, bC, cA, cB, cC\})$. This join is depicted in the figure below.

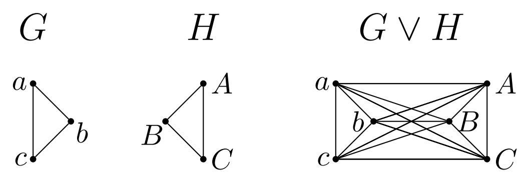

Figure 1.6: (left) G, H (middle), and $G \vee H$ (right).

Lastly, we define a few characteristics of graphs and their components. These may or may not be used frequently in this paper, but are important concepts to know in order to be able to talk about graphs comfortably.

Let G be a simple graph. We say two vertices $u, v \in V(G)$ are *adjacent* or *neighbors* if they form an edge $uv \in E(G)$. Similarly, we say a vertex is *incident* with an edge if it is one of its endpoints; $u \in V(G)$ is incident with $e \in E(G)$ if $e = uv$ for some $v \in V(G)$. The set of all vertices adjacent to v in G is called the *neighborhood* of v denoted $N_G(v)$ or simply $N(v)$. Sometimes this is refered to as the open neighborhood of v in G and then the closed neighborhood is defined via $N_G[v] = N_G(v) \sqcup \{v\}$. The *degree* of a vertex $v \in V(G)$ is the number of vertices adjacent to it and is denoted $deg_G(v)$ or simply $deg(v)$. Equivalently, the degree of a vertex v is the number of edges incident with it or the number of $|N_G(v)|$ neighbors that v has. We call a vertex of degree 1 a *leaf* and an edge incident with a leaf a *pendant* edge.

The following are three similar types of objects found in graphs.

Definition 1.1.7 (Walk)**.** Let G be a graph on n vertices. A *walk* in G is a sequence $(w_0, w_1, \ldots, w_k)$ of vertices in G whose adjacent elements must be adjacent in G. Adjacent elements in a walk must be distinct vertices but a vertex may be repeated multiple times throughout the sequence.

Definition 1.1.8 (Path)**.** Let G be a graph on n vertices. A *path* in G is a sequence $(v_0, v_1, \ldots, v_k)$ of distinct vertices in G whose adjacent elements must be adjacent in

G, and where no vertex is repeated throughout the sequence. This sequence gives the subgraph of G induced by $\{v_0v_1, v_1v_2, \ldots, v_{k-1}v_k\}$.

Definition 1.1.9 (Cycle)**.** Let G be a graph on n vertices. A *cycle* in G is a sequence $(v_0, v_1, \ldots, v_k, v_0)$ of internally distinct vertices (distinct except on the endpoints) that begins and terminates at the same vertex v_0. Often such a cycle is denoted via $(v_0v_1 \cdots v_k)$ and it is understood that the sequence wraps back around to v_0 after v_k. Additionally, the cycle $(v_0v_1 \cdots v_k)$ is equivalent to $(v_1 \cdots v_kv_0)$, $(v_2 \cdots v_kv_0v_1), \ldots$ and so on.

We call a simple graph G *acyclic* if it contains no cycles. If there exists a path from any vertex to every other vertex in G, then we call G *connected*. If not, we call G *disconnected*. We call the set of connected subgraphs of G whose disjoint union equals G the *connected components* of G.

This concludes the fundamental concepts needed to understand this project. The next and final section of this chapter will introduce all the fundamental families of graphs we refer to in subsequent chapters.

1.2 Fundamental Families of Graphs

In this section we introduce some fundamental families of graphs which we refer to throughout this paper. Often, instead of fully defining the graphs being worked with, we simply refer to it as a member of a larger family of graphs or as isomorphic to a family member. Some of these families overlap. It may be helpful to view a graph as a member of one family or another depending on the context.

Recall that a graph is acyclic if it contains no cycles. Similarly, we call a graph *k-cyclic* if it contains exactly k distinct cycles. If $k = 2$ or 3 we call it *bicyclic* or *tricyclic*, respectively. In a similar vein, we call a graph *k-partite* if we can partition its vertices into k disjoint sets such that no two vertices in the same partite set form an edge. If $k = 2$ or 3, we call it *bipartite* or *tripartite*, respectively. These are broad families of graphs often used to characterize subsets of graphs within another family. The following are some more important families of graphs.

Definition 1.2.1 (Complete Graph)**.** The *complete graph* on n vertices, denoted K_n, is the graph on n vertices such that every pair of distinct vertices shares an edge.

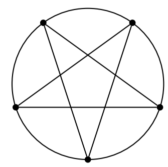

Figure 1.7: The Complete Graph K_5.

Definition 1.2.2 (Complete Bipartite Graph)**.** Let $m, n \in \mathbb{N}$. The *complete bipartite graph* $K_{m,n}$ is the bipartite graph whose vertices can be partitioned into two disjoint sets of sizes m and n, respectively, such that every vertex in the one partite set is adjacent to **every** vertex in the other partite set and no two vertices in the same partite set are adjacent.

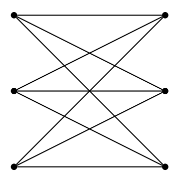

Figure 1.8: The Complete Bipartite Graph $K_{3,3}$.

Definition 1.2.3 (Complete Multipartite Graph)**.** The *complete k-partite graph* or *complete multipartite graph* $K_{n_1,\ldots,n_k}$ is the graph whose vertices can be partitioned into k disjoint sets of sizes n_1, n_2, $\ldots$, n_k, respectively such that every vertex in the one partite set is adjacent to **every** vertex in the other $k-1$ partite sets and no two vertices in the same partite set are adjacent.

If all partite sets are the same size n we call this graph the *complete equipartite graph* and denote it via $K_{n:k}$ or $K_{n\times k}$.

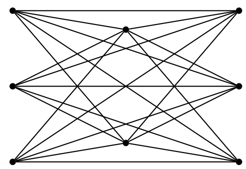

Figure 1.9: The Complete Multipartite Graph $K_{3,2,3}$.

Definition 1.2.4 (Cycle Graph)**.** The *cycle graph* on n vertices denoted C_n is a graph with exactly one distinct cycle containing all of its vertices.

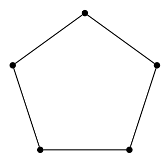

Figure 1.10: The Cycle Graph C_5.

Definition 1.2.5 (Tree)**.** A *tree* is any connected acyclic graph.

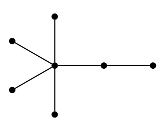

Figure 1.11: A tree on 7 vertices.

Definition 1.2.6 (Path Graph)**.** The *path* graph on n vertices, denoted P_n, is an acyclic graph with exactly one path containing all of its vertices.

Figure 1.12: The path P_4.

Definition 1.2.7 (Star Graph)**.** The *star graph* on $n + 1$ vertices, denoted $K_{1,n}$ (or S_{n+1} which we never use in this paper) consists of one central *hub* vertex adjacent to n

outer vertices, with no other edges. Sometimes this graph is refered to as an *n-star* or *n-edge star*.

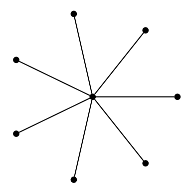

Figure 1.13: The 7-star $K_{1,7}$.

Definition 1.2.8 (Forest Graph)**.** Any disjoint union of tree graphs is called a *forest* graph.

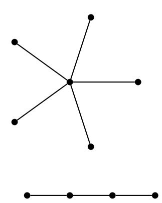

Figure 1.14: A forest on 10 vertices.

Definition 1.2.9 (Galaxy Graph)**.** Any disjoint union of star graphs is called a *galaxy* graph. We refer to $G = G_1 \sqcup \cdots \sqcup G_k$ as the $\{G_1, \ldots, G_k\}$*-galaxy* graph if $G_1, \ldots, G_k$ are all stars.

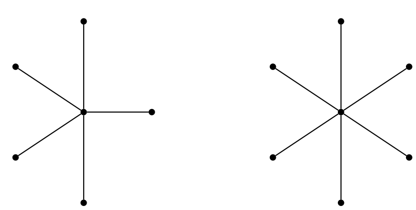

Figure 1.15: The $\{K_{1,5}, K_{1,6}\}$-Galaxy.

We have now defined a few important families of graphs which we will refer to throughout the rest of this paper. Again, we generally do not explicitly define every graph by its vertex and edge sets and simply refer to it as some member of a family or say that it is isomorphic to a member of a family. This is much more efficient and concise than listing out all vertices and edges as we did in the beginning of this chapter.

We are now ready to move on and introduce graph decompositions, the objects which are the subject of this project.

Chapter 2

Introduction

2.1 Decompositions

Suppose you have n translucent sheets of tracing paper with some points drawn on all n sheets of paper in the same set arrangement. Now, draw lines connecting points on each sheet of paper, so that no line appears on two distinct sheets of paper.

A graph K is depicted when all n sheets of tracing paper are aligned and stacked on top of each other with some light source present. Call the graph depicted on the ith sheet of paper G_i for $i = 1, \ldots, n$. The stacking of these sheets of paper depicts the edge-disjoint union $G_1 \cup \cdots \cup G_n = K$, and this collection of papers depicts the set $\{G_1, \ldots, G_n\}$ which we call a *graph decomposition* of K. This is defined formally below.

Definition 2.1.1 (Graph Decomposition)**.** Let K be a simple graph. We call a collection $\{G_1, \ldots, G_n\}$ of pairwise edge-disjoint subgraphs $G_1, \ldots, G_n \subseteq K$ of K a *decomposition* of K if their union equals K.

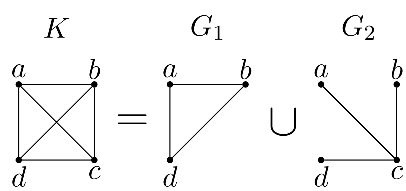

Figure 2.1: $\{G_1, G_2\}$ is a decomposition of K_4.

Graph decompositions are an important topic in combinatorics, graph theory, and design theory, with origins dating back to the 1800s. Notably, in 1850, Reverend Thomas Kirkman, a full-time clergyman and legendary mathematician, posed an important problem in *The Lady's and Gentleman's Diary* [8] now known as *the schoolgirl problem.* It goes

> *Fifteen young ladies in a school walk out three abreast for seven days in succession: it is required to arrange them daily so that no two shall walk twice abreast.*

The problem asks if we can form five distinct rows of three school girls on each day of the week so that no two school girls walk in the same row more than once in the week. This is equivalent to finding a decomposition of K_{15} whose members are all triangles. Both Kirkman and Arthur Cayley independently solved the schoolgirl problem and published their solutions in the 1851 edition of *The Lady's and Gentleman's Diary* [9]. Kirkman's solution is provided below.

Denoting the ladies by a_1, a_2, a_3; b_1, b_2, b_3; c_1, c_2, c_3; d_1, d_2, d_3; e_1, e_2, e_3, the following arrangement will be found to answer the question:

$$\begin{array}{c|c|c|c|c|c|c}
a_1\,a_2\,a_3 & a_1\,b_1\,c_1 & a_1\,d_1\,e_1 & a_1\,b_2\,d_2 & a_1\,c_2\,e_2 & a_1\,b_3\,e_3 & a_1\,c_3\,d_3 \\
b_1\,b_2\,b_3 & a_2\,b_2\,c_2 & a_2\,d_2\,e_2 & a_2\,b_3\,d_3 & a_2\,c_3\,e_3 & a_2\,b_1\,e_1 & a_2\,c_1\,d_1 \\
c_1\,c_2\,c_3 & a_3\,d_3\,e_3 & a_3\,b_3\,c_3 & a_3\,c_1\,e_1 & a_3\,b_1\,d_1 & a_3\,c_2\,d_2 & a_3\,b_2\,e_2 \\
d_1\,d_2\,d_3 & b_3\,d_1\,e_2 & b_1\,c_1\,e_3 & b_1\,c_3\,e_1 & b_2\,c_3\,d_1 & c_2\,b_3\,e_1 & c_2\,b_3\,e_1 \\
e_1\,e_2\,e_3 & c_3\,d_2\,e_1 & e_3\,b_2\,c_1 & d_1\,c_2\,e_3 & c_1\,d_2\,b_3 & d_2\,b_1\,c_2 & c_1\,d_3\,b_2
\end{array}$$

This is the symmetrical and only possible solution. All others differ from this only in disturbing the alphabetical order, or that of the three subindices in certain triplets of the first column, or in both these together.

Each triple in the array above gives an edge-distinct triangle subgraph of K_{15} whose vertex set we take to be $\{a_1, a_2, a_3, \dots e_1, e_2, e_3\}$. The set of all these subgraphs is a decomposition of K_{15}. Since all of these subgraphs are isomorphic to C_3, we call it a C_3*-decomposition.* This is a special type of decomposition which is defined formally on the following page.

Definition 2.1.2 (G-decomposition)**.** A *G-decomposition* of a graph K is a decomposition $\{G_1, \ldots, G_t\}$ of K whose members are all isomorphic to some graph G. If such a set exists we say that K *allows* a G-decomposition or equivalently, that G *decomposes* K. If $K \cong K_n$ we sometimes call the decomposition a *G-design of order n*.

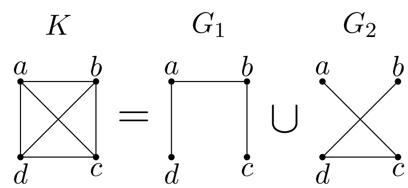

Figure 2.2: $\{G_1, G_2\}$ is a P_3-decomposition of K_4 or a P_3-design of order 4.

We know that if a G-decomposition of some graph K exists, then all of its members have the same number of edges and vertices. This allows us to find constraints on graphs K that can be decomposed by some subgraph $G \subseteq K$ on m edges solely based on divisiblity properties.

Lemma 2.1.3 (Necessary Condition (general))**.** *Let G be a simple graph on m edges. There exists a G-decomposition of a graph K only if m divides $|E(K)|$.*

Proof. Suppose there exists a G-decomposition $\{G_1, \ldots, G_n\}$ of K. Then $E(G_1) \sqcup \cdots \sqcup E(G_t) = E(K)$ and so $|E(K)| = |E(G_1) \sqcup \cdots \sqcup E(G_t)| = |E(G_1)| + \cdots + |E(G_t)| = tm$. So $|E(G)| = m$ divides $|E(K)|$. □

Theorem 2.1.4 (Necessary Condition (K_n))**.** *Let G be a simple graph on m edges. There exists a G-decomposition of K_n only if n is idempotent modulo $2m$, that is, only if $n^2 \equiv n \pmod{2m}$.*

Proof. Suppose there exists a G-decomposition of K_n. Then $|E(G)| = m$ divides $|E(K_n)| = \binom{n}{2} = \frac{n(n-1)}{2}$ by Lemma 2.1.3. Therefore, $\frac{n^2-n}{2} = mt$ for some $t \in \mathbb{N}$. Observe

$$n^2 - n = 2mt \implies n^2 - n \equiv 0 \pmod{2m} \implies n^2 \equiv n \pmod{2m}.$$

□

By the previous theorem, any graph on m edges decomposes K_n only if n is idempotent modulo $2m$. Note that the converse is not necessarily true. However, for a graph G on m edges, this finite set of constraints allows us to ask:

For what n is K_n G-decomposable?

This question is known as the *spectrum problem* for graph decompositions. Pioneering work by Rosa and Kotzig in the 1960s especially in the development of graph labelings helped shape the modern approach to G-decomposition problems. Since then, labeling-based techniques and tools from design theory have driven significant progress. In particular, graph labeling methods have played a central role in addressing the spectrum problem for small graphs. This thesis project directly builds upon work by Freyberg and Peters, who recently solved the spectrum problem for forests with six edges [4]. Their paper provides a comprehensive summary of known decompositions for graphs G with fewer than seven edges.

Using graph labelings to solve G-decomposition problems is basically about doing algebra on subgraphs in order to generate other edge-disjoint subgraphs while preserving the structure of G. If we take the vertices of a graph K to be elements of a group, we can use the structure of the group to our advantage. Specifically, when $K \cong K_n$, and we take it's vertices to be $\mathbb{Z}_n$, and then we label the vertices of G with some subset of $\mathbb{Z}_n$. There are various labeling techniques of this kind stemming from Rosa's work in the 1960s that allow us permute or act on the labels of the vertices of G with subgroups of $\mathbb{Z}_n$ to generate new isomorphic copies of G that are pairwise edge-disjoint. In the next section, we provide an example which outlines in some detail how this machinery works for G-decompositions of complete graphs.

2.2 Graph labelings

Take the vertices of K_5 to be $\mathbb{Z}_5$ and arrange them in the same manner as in Figure 1.7. Notice that every vertex shares an edge with two vertices directly adjacent to it and two vertices that are 'two adjacencies away' on the outer cycle (01234). We say two vertices u, v have *length* $\ell(uv) = l$ if they are 'l adjacencies away' from each other on the outer cycle.

Formally, for K_n we define the edge length function ℓ as follows:

$$\ell(uv) = \min\{|u - v|, n - |u - v|\}.$$

Notice that for K_5, we only have lengths 1 or 2 as previously observed. Color the length 1 edges blue, and the length 2 edges red. This is depicted in the figure below.

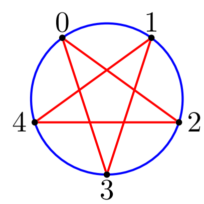

Figure 2.3: K_5 with lengths colored.

Now, consider P_3. It has 2 edges, and K_5 has $\binom{5}{2} = 10$ edges. Since $2|10$, by Lemma 2.1.3 it's *possible* that a P_3-decomposition of K_5 exists. Now since P_3 has 2 edges and there are 2 lengths in K_5, now we can just make sure each copy of P_3 has both a blue edge and a red edge. How can we do that while ensuring that no edge is repeated?

It turns out that if we take the vertices of K_n to be $\mathbb{Z}_n$, adding 1 (and therefore anything) modulo n to the endpoints of an edge preserves its length. We call the act of permuting vertices in this manner (permuting by an element of the group) *clicking* or *developing*. If we are permuting all vertices of a labeling by the same group element, we often will just say we are developing the labeling by that group element.

In the context of our problem with P_3 and K_5, this means that if we label P_3 with elements of $\mathbb{Z}_5$ such that we have one blue edge of length 1, and one red edge of length 2, we can generate all edges of length 1 and 2 in K_n (which is all edges of K_n) while preserving the structure of the graph by developing our labeling by 1. Label the defining path of P_3 via (2,0,1). Developing the vertices by 1 (modulo 5) will give all members of a P_3-decomposition of K_5. A decomposition that can be generated by permuting all vertices of one labeling repeatedly in this fashion is called a *cyclic decomposition*. This is depicted in the following figure.

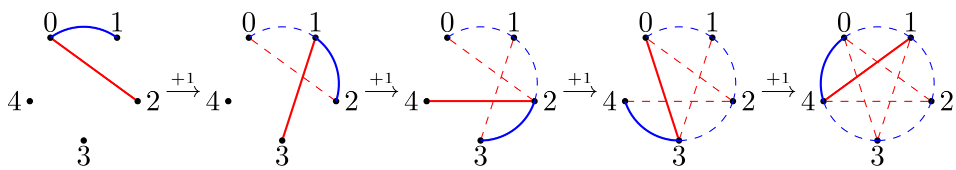

Figure 2.4: A cyclic P_3-decomposition of K_5.

Nice and easy right? But that is just one complete graph that P_3 can decompose. Remember, that it is *possible* that P_3 can decompose any K_n where $n \equiv n^2 \pmod 4$ by Theorem 2.1.4. This equivalent to saying $n = 4t+r$ where r is an idempotent in the ring $\mathbb{Z}_4$ and $t \geq 1$. The idempotents in $\mathbb{Z}_4$ are $0, 1$. So this means $\mathbb{Z}_5$ is just a special case of n where $n = 4t+1$ and $t = 1$. Even though these are infinite families, it is known that for each step $t \mapsto t+1$, new lengths come 2 at a time. This means if we can somehow transform our labeling at each step to include the new lengths, we can decompose the entire family K_{4t+1}. We want to fine tune our labeling to achieve this. Note that if r is not 0 or 1, we need multiple labelings to decompose K_n. This is explained later in this paper.

Lastly, some basic observations about a general subgraph $G \subseteq K_n$ with m edges of K_n. The maximal length in K_n is $\lfloor \frac{n}{2} \rfloor$. This is intuitive, since when you travel halfway across the outer cycle from some vertex, the lengths start going back down again as you begin nearing that vertex again. Now, n must be of the form $2mt + r$ where $t \geq 1$ and r is an idempotent in the ring $\mathbb{Z}_{2m}$. This means that in K_{2m+r} if $\ell(uv) = |u - v| < \lfloor \frac{2m+r}{2} \rfloor < \lfloor \frac{2mt+r}{2} \rfloor$ for $t > 1$, then $\ell(uv) = |u - v|$ in all K_{2mt+r} for $t \geq 1$. This is important, because at each step $t \mapsto t+1$, new lengths come m at a time.

Now, for $r = 0$ or 1, if a certain labeling of a graph G on m edges exists, there exists a G-decomposition of K_{2mt+r} for $t \geq 1$. However, if $r \neq 0, 1$, one labeling will not suffice and other techniques are needed to prove that G decomposes K_{2mt+r} for $t \geq 1$. These labelings and techniques are defined as they are needed in the following chapters. Finally, we are ready to introduce the focus of this project.

2.3 Seven edge forests

This project continues on Freyberg and Peters' work on six edge forests by asking the same question about seven edge forests:

> Let F be a forest on seven edges. For which n does F decompose K_n?

Every component of a forest on seven edges is a tree on six or less edges which are cataloged in Figure 2.5. We use the naming convention $\mathbf{T_j^i}$ to denote the i^{th} tree with j vertices and we index the vertices v_1 through v_j for each tree as specified below.

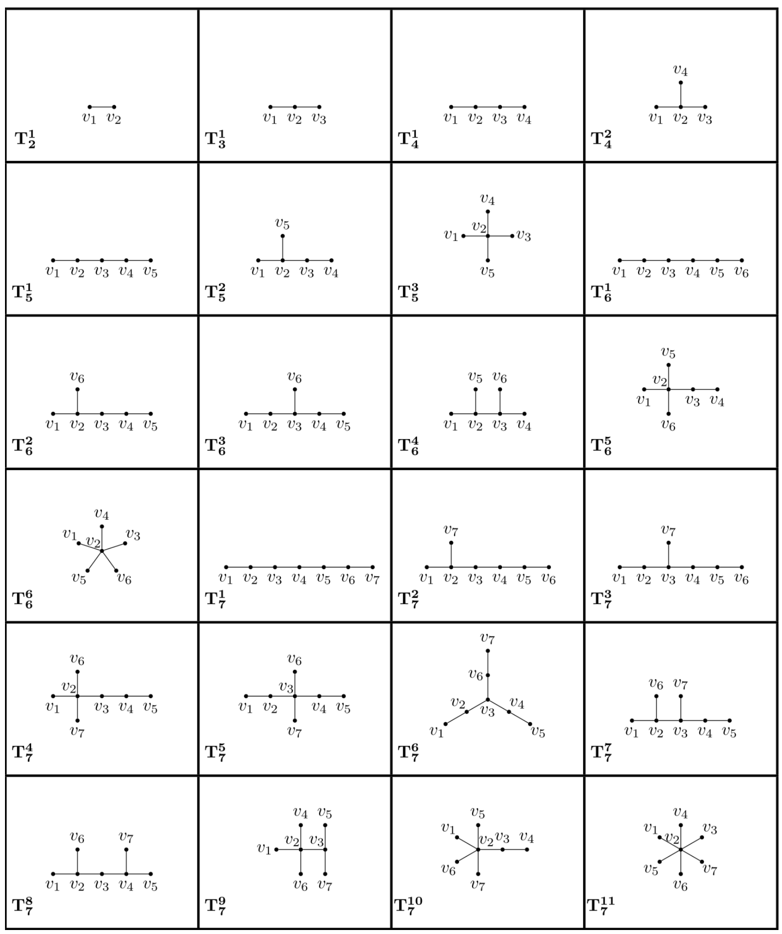

Figure 2.5: Trees with less than seven edges.

The next theorem gives the necessary conditions for the existence of a G-decomposition of K_n when G is a graph with 7 edges.

Theorem 2.3.1. *If G is a graph with 7 edges and a G-decomposition of K_n exists, then $n \equiv 0, 1, 7,$ or $8 \pmod{14}$.*

Proof. If a G-decomposition of K_n exists, then n is idempotent modulo $2(7) = 14$ by Theorem 2.1.4 which immediately implies that $n \equiv 0, 1, 7, \text{or } 8 \pmod{14}$ since those are all the idempotents in $\mathbb{Z}_{14}$. □

For this project, we do not define the graph on one vertex to be a tree. This means that any connected component in a forest has at least one edge and we also require there to be at least two connected components. There are 47 such forests with 7 edges up to isomorphism. The spectrum problem for the matching $\mathbf{7T_2^1}$ was solved by de Werra in 1970, so only the remaining 46 forests need be considered in the subsequent chapters. Chapter 3 handles decomposing K_n into all 47 forests when $n \equiv 0$ or 1 $\pmod{14}$. Chapter 4 applies to all the forests when $n \equiv 7$ or $8 \pmod{14}$ with the lone exception of $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$, which is solved for those values of n in Chapter 5.

After proving the main result of this project, we provide two additional results in Chapter 6: (1) An edge mapping that depends on t and preserves lengths for wraparound edges in K_{3m} and K_{3m+1} to K_{2mt+m} and $K_{2mt+m+1}$, respectively, for each step $t \mapsto t+1$. (2) Galaxy decompositions of complete bipartite graphs. Chapter 6 concludes the main results of this thesis project.

We then present some `python` programs in Chapter 7 that were created as a result of this thesis project. One is `tikzgrapher`: A new graph visualization program, which was built from scratch only using `pygame` and `NetworkX` and several other basic libraries. It allows one to visualize simple `NetworkX` graphs in an interactive `pygame` window that allows for colorings and custom labelings along with dragging and moving components of the graph. Moreover, the user can save the graph layout depicted in the `pygame` window as a `TikZ` graph in a standalone LaTeX file. `tikzgrapher` is paired with a graph labeling solver. This is a constraint programming project that can find several labelings on graphs if they exist. A conclusion follows Chapter 7, then a list of references and an appendix of labelings marks the end of the thesis.

Chapter 3

$\mathbf{n \equiv 0, 1 \pmod{14}}$

To begin this chapter, we extend intuition developed in the Chapter 2 to present some machinery specific to K_n where $n \equiv 0, 1 \pmod{14}$. This informs the formal definitions and theorems we use for this case.

3.1 Construction for $n \equiv 0, 1 \pmod{14}$

K_0 and K_1 do not have enough vertices to contain a forest on seven edges. So K_{14} and K_{15} are the base graphs for K_n where $n \equiv 0$ or $1 \pmod{14}$, respectively. We first show how to decompose K_{14} and K_{15} in Subsection 3.1.1 and then show how to generalize this to their respective families in Subsection 3.1.2

To be absolutely clear, the concepts found in this section are **not our own original ideas**. Rosa and his colleagues, developed these labeling ideas decades ago. Other mathematicians since also have helped in developing labeling techniques. This is just an explanation from our perspective, of the amazing work of Rosa and his successors in creating graph labelings techniques to solve decomposition problems.

3.1.1 K_{14} and K_{15}

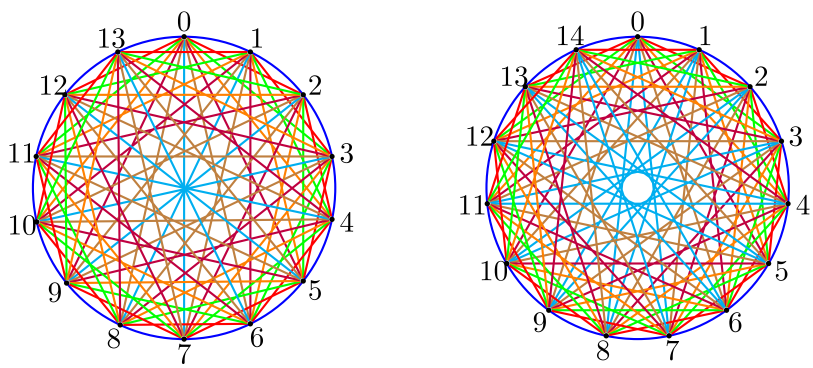

Figure 3.1: K_{14} (left) and K_{15} (right) with edges colored by length.

Both K_{14} and K_{15} only have edges with lengths 1 through 7, however, there is important distinction in counting edges of one length in K_{14} versus counting them in K_{15}. The cyan edges have length 7, and convieniently, because of the way these graphs are drawn in the figure above, we can easily count them as they form the innermost spoke of the graphs. Because these graphs are so similar, we can use one labeling to deal with both at the same time! However, we must be very careful in doing so.

In K_{14} there are only 7 edges of length 7 and 14 edges of all other lengths, yet in K_{15} there are 15 edges of length 7 along with 15 edges of all other lengths. It turns out this pattern generalizes. In K_n,

If n is odd: There are n edges of lengths 1 through $\frac{n-1}{2}$.

If n is even: There are n edges of lengths 1 up to $\frac{n}{2}$, and then $\frac{n}{2}$ edges of length $\frac{n}{2}$. This is why the labeling of P_3 in Figure 2.4 used to decompose K_5 in the Chapter 2 worked so easily. Because K_5 has odd order, there are 5 edges of each length in $\{1, 2\}$. The same applies here for K_{15}, we just label each seven edge forest F so that they contain all seven lengths in K_{15}, and then develop the labelings by 1 to generate the entire F-decomposition of K_{15}.

But notice that if we develop the same labeling of F in K_{14} (assuming we do not use the vertex 14) we would overcount length 7 edges. There is a simple remedy for this, but it requires a shift in perspective.

Take $V(K_{14})$ to be $\mathbb{Z}_{13} \cup \{\infty\}$, and label all edges via the length function modulo

13 except for edges incident to ∞ which will refer to as length ∞. We do this so that, intuitively, developing vertices by 1 fixes the ∞ vertex so that $\infty \mapsto \infty + 1 = \infty$. Formally:

$$\ell(uv) = \begin{cases} \min\{|u-v|, 13-|u-v|\}, & u, v \neq \infty, \\ \infty, & u \text{ or } v = \infty \end{cases} \quad \text{and } v \mapsto \begin{cases} v+1, & v \in \mathbb{Z}_{13}, \\ \infty, & v = \infty. \end{cases}$$

Another reason for doing this is that we will now have 13 edges of lengths 1 through 6 as well as of length ∞, since the ∞ vertex will have all 13 edges of length ∞ to vertices in $\mathbb{Z}_{13}$ adjacent to it. Now, if we develop the endpoints of an edge with any length by 1 repeatedly, we will get all 13 distinct edges of that length. So we can cyclically generate F-decompositions with this new construction for K_{14} for each forest.

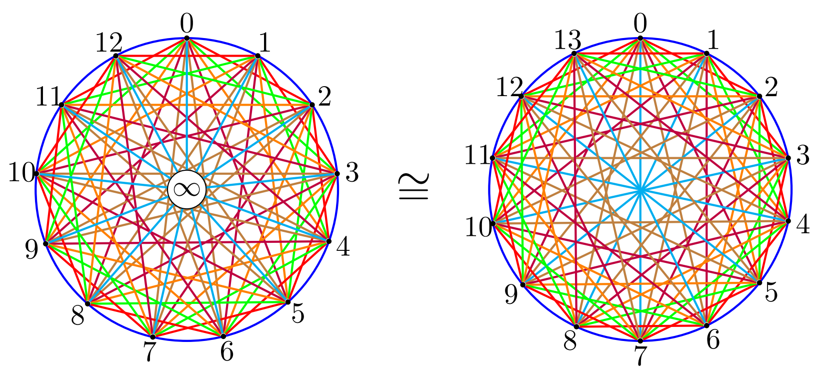

Figure 3.2: $K_{13} \vee K_1$ is isomorphic to K_{14}.

It is important to note that we **must** use edges uv of length $|u-v|$ in K_{13} so they will also have length $|u-v|$ in K_{14} and K_{15} if we want to use one labeling for both cases. We also must ensure that the edge of length 7 is a pendant edge, so that we can relabel a leaf incident with that edge as ∞ without disturbing other edge lengths.

Putting this all together: if we can just use vertex labels from $\mathbb{Z}_{13}$ for a labeling of a forest F but only use edges uv of lengths $|u-v| = l$ modulo 14 for each length $l \in \{1, \ldots, 7\}$, ensuring that the edge of length 7 is a pendant edge, we can simply develop the labeling by 1 to get the F-decomposition of K_{15}, relabel a leaf incident with the length seven edge with ∞, and then once more develop by 1 to get the F-decomposition of K_{14}. We show an example on the next page.

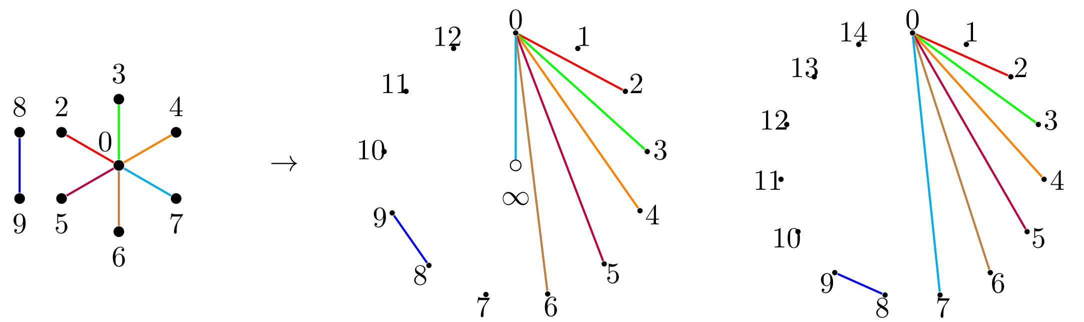

Figure 3.3: A labeling (left) that gives the $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{11}}$-decomposition of K_{14} (middle) and K_{15} (right), respectively, when developed by 1. The leaf 7 in the pendant edge $(0, 7)$ (of length 7 modulo 14) is relabeled as ∞ for K_{14}.

3.1.2 Stretching a labeling and generalizing to K_{14t} and K_{14t+1} for $t > 1$

Now, that we have learned how to decompose K_{14} and K_{15}, what about the rest of the family? Well, K_{14t} and K_{14t+1} actually contain edges of lengths in

$$\bigsqcup_{0 \leq i < t} \{1 + 7i, \ldots, 7 + 7i\} \text{ for } t \geq 1.$$

This makes sense because the maximal length in K_{14t} and K_{14t+1} is $\lfloor \frac{14t+1}{2} \rfloor = 7t$. It turns out that we can actually just replicate the process we did for lengths 1 through 7 for each step $t \mapsto t+1$ for K_{14t} and K_{14t+1} for the new lengths that come 7 at a time if we add some more restrictions.

There are two types of edges in complete graphs. In K_n, we call edges uv of length $n - |u - v|$ *wraparound edges*, and edges ab of length $|a - b|$ *short edges*.

In K_{14}: $\ell((0, 8)) = \min\{|0 - 8|, 14 - |0 - 8|\} = \min\{8, 6\} = 6$.

In K_{28}: $\ell((0, 8)) = \min\{|0 - 8|, 28 - |0 - 8|\} = \min\{8, 20\} = 8$.

For example, $(0, 8)$ is a wraparound edge in K_{14} but a short edge in K_{28}. If we want to build a labeling that will generalize to an entire family of complete graphs, we need to understand how these types of edges would generalize as the order of the complete graph increases.

There is a way to map wraparound edges at each step to preserve length which is described in Chapter 6, but it can get ugly. However, there is a way to avoid this problem entirely. If we only use *short edges* uv that have length $|u - v|$ in K_{14} for our labelings, then that edge will have length $|u - v|$ in every complete graph K_n where $n > 14$. We show another example below for $t > 2$.

In $K_{14\cdot 1}$: $\ell((0, 6)) = \min\{|0 - 6|, 14 - |0 - 6|\} = \min\{6, 8\} = 6$.

In K_{28}: $\ell((0, 6)) = \min\{|0 - 6|, 28 - |0 - 6|\} = \min\{6, 22\} = 6$.

$$\vdots$$

In $K_{14\cdot t}$: $\ell((0, 6)) = \min\{|0 - 6|, 14t - |0 - 6|\} = \min\{1, 22 + 14(t - 2)\} = 6$.

So we see that if we only use short edges, the length of edges in our labeling will be preserved as we scale up. So, for that reason, we use them. However, another important feature is that if we only use short edges uv, we know that without loss of generality $v > u$ and so $|u - v| = v - u$. This introduces another extremely important mechanism we can exploit to generalize labelings.

If we bundle the short edge requirement and the requirement that the maximal edge is a pendant edge with a new ordered bipartition $V(G) = A \sqcup B$ requirement on the vertices of a labeling G of F such that all vertices $a \in A$ only have neighbors in B that are larger than a, then we get a really nice property. All edges are now of the form ab where $a \in A$ and $b \in B$ so that $a < b$ and so all such edges ab have length $b - a$ modulo 14. This means that if $\ell(ab) = l$ in K_{14}, then $b - a = l$ and so $(b + c) - a = l + c$. More importantly, if we then add any $c \in \mathbb{Z}_{14t}$ to all vertices in the larger partite set B of our labeling, we will simply increase the lengths of all edges in our labeling by c in $\mathbb{Z}_{14t}$ and $\mathbb{Z}_{14t+1}$ for $t \geq 1$ as a result of bundling these restrictions together! We call the act of adding some constant c to all vertices of the larger partite set B in a labeling of this type *stretching*.

Let $t > 1$, and consider K_{14t+1}. If we label a seven edge graph F to contain lengths 1 through 7 modulo 14 and adhere to the restrictions summarized above, developing by 1 will generate all edges of lengths in 1 through 7 in K_{14t+1}. Then, if we stretch that labeling by $7i$ for each $0 < i < t$ and develop by 1 repeatedly at each step, we will generate all edges of the remaining lengths in K_{14t+1} and therefore generate all edges in K_{14t+1} in isomorphic copies of F to get an F-design of order $14t + 1$.

Now, in the same labeling of F in K_{14t}, recall that there are only $7t$ edges of the maximal length $7t$. This means that we want to take its vertices to be $\mathbb{Z}_{14t-1} \cup \{\infty\}$ and so there are now $14t - 1$ edges of length 7 along with all lengths less than $7t$. T we simply develop G by 1 to generate lengths in $\{1, \ldots, 7\}$, then stretch that labeling by $7i$ and develop by 1 for each $0 < i < t$ **except in the last labeling**, which was stretched by $7(t-1)$ with lengths in $\{7t-6, \ldots, 7t\}$. Since the pendant edge of length 7 was stretched to be one of maximal length $7t$, it is still a pendant edge. So now we relabel the leaf as ∞ and so it's length becomes ∞. We then develop it by 1 and have collected all edges in K_{14t} while generating an F-design of order $14t$.

We show how the labeling from Figure 3.3 gives the $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{11}}$-decomposition of $K_{14(2)} = K_{28}$ and $K_{14(2)+1} = K_{29}$ below.

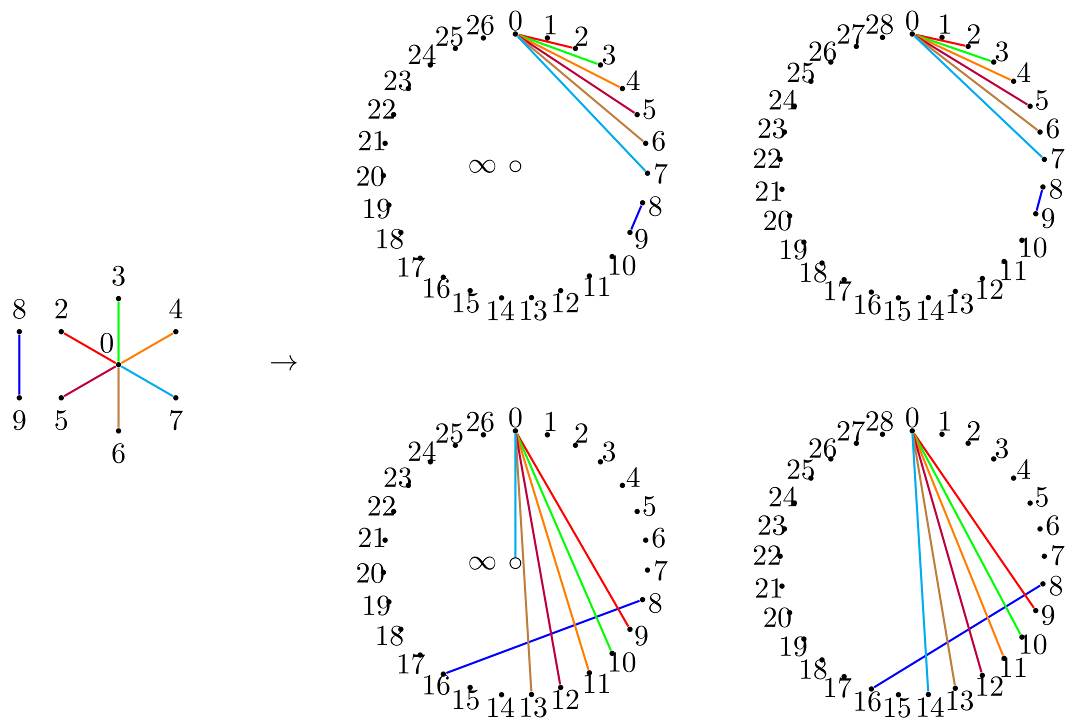

Figure 3.4: A single labeling (left) gives four labelings which when developed by 1 and stretched by 7 give the $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{11}}$-decomposition of (middle) K_{28} and (right) K_{29}.

We will refer to this act of stretching a labeling for the next construction, but first we put the ideas in this section together formally in the next section.

3.2 Results for $n \equiv 0, 1 \pmod{14}$

We begin with Rosa's original labeling of a bipartite graph G on m edges which uses lengths modulo $2m$ (in K_{2m+1}).

Definition 3.2.1 ((Rosa [7])). Let G be a graph with m edges. A *ρ-labeling* of G is an injection $f : V(G) \to \{0, 1, 2, \ldots, 2m\}$ that induces a bijective *length function* $\ell : E(G) \to \{1, 2, \ldots, m\}$ where

$$\ell(uv) = \min\{|f(u) - f(v)|, 2m + 1 - |f(u) - f(v)|\},$$

for all $uv \in E(G)$.

Rosa showed that if a ρ-labeling of a graph G with m edges exists, then a cyclic G-decomposition of K_{2m+1} exists, and the converse also holds. This is the next theorem. Later, Rosa and others began studying more restrictive types of ρ-labelings to decompose more complete graphs.

Next, we define some of these labelings and theorems associated with them.

Theorem 3.2.2 (Rosa [7]). *Let G be a graph with m edges. There exists a cyclic G-decomposition of K_{2m+1} if and only if G admits a ρ-labeling.*

Definition 3.2.3 (Rosa [7]). A σ-labeling of a graph G is a ρ-labeling such that $\ell(uv) = |f(u) - f(v)|$ for all $uv \in E(G)$.

Definition 3.2.4 (El-Zanati, Vanden Eynden [2]). A ρ- or σ-labeling of a bipartite graph G with bipartition (A, B) is called an *ordered* ρ- or σ-labeling and denoted ρ^+, σ^+, respectively, if $f(a) < f(b)$ for each edge ab with $a \in A$ and $b \in B$.

Theorem 3.2.5 (El-Zanati, Vanden Eynden [2]). *Let G be a graph with m edges which has a ρ^+-labeling. Then, G decomposes K_{2mk+1} for all positive integers k.*

Definition 3.2.6 (Freyberg, Tran [5]). A *σ^{+-}-labeling* of a bipartite graph G with m edges and bipartition (A, B) is a σ^+-labeling with the property that $f(a) - f(b) \neq m$ for all $a \in A$ and $b \in B$, and $f(v) \notin \{2m, 2m - 1\}$ for any $v \in V(G)$.

Theorem 3.2.7 (Freyberg, Tran [5])**.** *Let G be a graph with m edges and a σ^{+-}-labeling such that the edge of length m is a pendant. Then there exists a G-decomposition of both K_{2mk} and K_{2mk+1} for every positive integer k.*

Table 3.1 gives σ^{+-}-labelings of all forests on 7 edges except the matching. The vertex labels of each connected component with k vertices are given as a k-tuple, $(v_1, \ldots, v_k)$ corresponding to the vertices $v_1, \ldots, v_k$ positioned as shown in Figure 2.5. We leave it to the reader to infer the bipartition (A, B).

Example 3.2.8. *A σ^{+-}-labeling of $\mathbf{T_6^6} \sqcup 2\mathbf{T_2^1}$ is shown in Figure 3.5. The vertices labeled $1, 2$ and 9 belong to A, and the others belong to B. The lengths of each edge are indicated on the edge.*

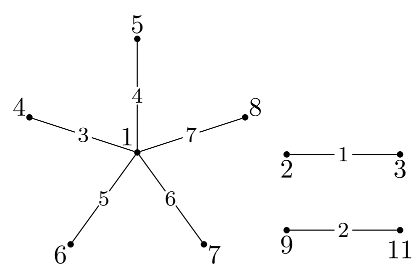

Figure 3.5: σ^{+-}-labeling of $\mathbf{T_6^6} \sqcup 2\mathbf{T_2^1}$.

The labelings given in Table 3.1 along with Theorem 3.2.7 are enough to conclude this case.

Theorem 3.2.9. *Let F be a forest with 7 edges. There exists an F-decomposition of K_n whenever $n \equiv 0$ or $1 \pmod{14}$.*

Proof. The matching $7\mathbf{T_2^1}$ has been solved by de Werra in [10]. The rest of the proof follows from Theorem 3.2.7 and the labelings given in Table 3.1.

□

Table 3.1: σ^{+-}-labelings for forests with seven edges.

Forest	Vertex Labels
$\mathbf{T_7^1} \sqcup \mathbf{T_2^1}$	$(0, 6, 1, 5, 2, 9, 7) \sqcup (3, 4)$
$\mathbf{T_7^3} \sqcup \mathbf{T_2^1}$	$(9, 2, 5, 1, 6, 0, 3) \sqcup (8, 7)$
$\mathbf{T_7^2} \sqcup \mathbf{T_2^1}$	$(9, 2, 5, 1, 6, 0, 4) \sqcup (8, 7)$
$\mathbf{T_7^4} \sqcup \mathbf{T_2^1}$	$(5, 1, 4, 2, 9, 6, 7) \sqcup (10, 11)$
$\mathbf{T_7^5} \sqcup \mathbf{T_2^1}$	$(3, 8, 1, 4, 2, 5, 7) \sqcup (9, 10)$
$\mathbf{T_7^8} \sqcup \mathbf{T_2^1}$	$(7, 8, 1, 6, 0, 4, 3) \sqcup (9, 11)$
$\mathbf{T_7^9} \sqcup \mathbf{T_2^1}$	$(8, 1, 6, 3, 4, 5, 7) \sqcup (9, 10)$
$\mathbf{T_7^{10}} \sqcup \mathbf{T_2^1}$	$(6, 1, 5, 3, 8, 4, 7) \sqcup (9, 10)$
$\mathbf{T_7^6} \sqcup \mathbf{T_2^1}$	$(5, 11, 9, 10, 6, 12, 7) \sqcup (8, 1)$
$\mathbf{T_7^7} \sqcup \mathbf{T_2^1}$	$(4, 8, 1, 6, 0, 5, 3) \sqcup (9, 10)$
$\mathbf{T_6^1} \sqcup \mathbf{T_3^1}$	$(0, 6, 1, 5, 2, 9) \sqcup (11, 10, 12)$
$\mathbf{T_6^2} \sqcup \mathbf{T_3^1}$	$(3, 6, 1, 8, 4, 0) \sqcup (10, 9, 11)$
$\mathbf{T_6^3} \sqcup \mathbf{T_3^1}$	$(5, 11, 9, 12, 7, 10) \sqcup (1, 8, 4)$
$\mathbf{T_6^4} \sqcup \mathbf{T_3^1}$	$(3, 8, 4, 1, 6, 7) \sqcup (10, 9, 11)$
$\mathbf{T_6^5} \sqcup \mathbf{T_3^1}$	$(5, 1, 8, 3, 4, 7) \sqcup (10, 9, 11)$
$\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$	$(4, 1, 8, 5, 6, 7) \sqcup (10, 9, 11)$
$\mathbf{T_5^1} \sqcup \mathbf{T_4^1}$	$(0, 6, 1, 5, 2) \sqcup (9, 8, 10, 3)$
$\mathbf{T_5^2} \sqcup \mathbf{T_4^1}$	$(7, 1, 8, 5, 6) \sqcup (0, 4, 2, 3)$
$\mathbf{T_5^2} \sqcup \mathbf{T_4^2}$	$(7, 1, 8, 4, 6) \sqcup (10, 9, 11, 12)$
$\mathbf{T_5^3} \sqcup \mathbf{T_4^1}$	$(6, 0, 3, 4, 5) \sqcup (8, 7, 9, 2)$

Table 3.1: σ^{+-}-labelings for forests with seven edges.

Forest	Vertex Labels
$\mathbf{T_5^1} \sqcup \mathbf{T_4^2}$	$(4, 8, 1, 7, 2) \sqcup (10, 9, 11, 12)$
$\mathbf{T_5^3} \sqcup \mathbf{T_4^2}$	$(6, 0, 3, 4, 5) \sqcup (8, 9, 2, 7)$
$\mathbf{T_6^1} \sqcup 2\mathbf{T_2^1}$	$(0, 6, 1, 5, 2, 9) \sqcup (8, 10) \sqcup (3, 4)$
$\mathbf{T_6^2} \sqcup 2\mathbf{T_2^1}$	$(3, 6, 1, 8, 4, 0) \sqcup (5, 7) \sqcup (9, 10)$
$\mathbf{T_6^5} \sqcup 2\mathbf{T_2^1}$	$(4, 1, 8, 3, 5, 7) \sqcup (0, 2) \sqcup (9, 10)$
$\mathbf{T_6^4} \sqcup 2\mathbf{T_2^1}$	$(5, 8, 4, 1, 6, 7) \sqcup (0, 2) \sqcup (9, 10)$
$\mathbf{T_6^3} \sqcup 2\mathbf{T_2^1}$	$(5, 11, 9, 12, 7, 10) \sqcup (8, 1) \sqcup (0, 4)$
$\mathbf{T_6^6} \sqcup 2\mathbf{T_2^1}$	$(4, 1, 8, 5, 6, 7) \sqcup (2, 3) \sqcup (9, 11)$
$\mathbf{T_5^1} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(0, 6, 1, 5, 2) \sqcup (8, 10, 9) \sqcup (11, 4)$
$\mathbf{T_5^2} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(7, 1, 8, 5, 6) \sqcup (10, 9, 11) \sqcup (0, 4)$
$\mathbf{T_5^3} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(6, 0, 3, 4, 5) \sqcup (1, 8, 7) \sqcup (9, 11)$
$2\mathbf{T_4^1} \sqcup \mathbf{T_2^1}$	$(0, 6, 1, 5) \sqcup (2, 9, 7, 10) \sqcup (3, 4)$
$\mathbf{T_4^1} \sqcup \mathbf{T_4^2} \sqcup \mathbf{T_2^1}$	$(11, 9, 10, 7) \sqcup (4, 0, 5, 6) \sqcup (8, 1)$
$2\mathbf{T_4^2} \sqcup \mathbf{T_2^1}$	$(4, 0, 5, 6) \sqcup (10, 9, 11, 12) \sqcup (8, 1)$
$\mathbf{T_4^1} \sqcup 2\mathbf{T_3^1}$	$(0, 6, 1, 5) \sqcup (8, 10, 9) \sqcup (11, 4, 7)$
$\mathbf{T_4^2} \sqcup 2\mathbf{T_3^1}$	$(4, 0, 5, 6) \sqcup (1, 8, 7) \sqcup (11, 9, 12)$
$\mathbf{T_4^1} \sqcup \mathbf{T_3^1} \sqcup 2\mathbf{T_2^1}$	$(0, 6, 1, 5) \sqcup (8, 10, 7) \sqcup (11, 4) \sqcup (2, 3)$
$\mathbf{T_4^2} \sqcup \mathbf{T_3^1} \sqcup 2\mathbf{T_2^1}$	$(4, 0, 5, 6) \sqcup (11, 9, 12) \sqcup (2, 3) \sqcup (8, 1)$
$\mathbf{T_5^1} \sqcup 3\mathbf{T_2^1}$	$(0, 6, 1, 5, 2) \sqcup (10, 3) \sqcup (9, 7) \sqcup (11, 12)$
$\mathbf{T_5^2} \sqcup 3\mathbf{T_2^1}$	$(6, 1, 8, 4, 7) \sqcup (3, 5) \sqcup (9, 12) \sqcup (10, 11)$
$\mathbf{T_5^3} \sqcup 3\mathbf{T_2^1}$	$(3, 0, 4, 5, 6) \sqcup (8, 1) \sqcup (10, 11) \sqcup (9, 7)$
$3\mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(0, 6, 1) \sqcup (4, 8, 5) \sqcup (2, 9, 7) \sqcup (10, 11)$
$\mathbf{T_4^1} \sqcup 4\mathbf{T_2^1}$	$(0, 6, 1, 5) \sqcup (9, 2) \sqcup (8, 10) \sqcup (4, 7) \sqcup (11, 12)$
$\mathbf{T_4^2} \sqcup 4\mathbf{T_2^1}$	$(4, 0, 5, 6) \sqcup (2, 3) \sqcup (9, 11) \sqcup (8, 1) \sqcup (10, 7)$
$2\mathbf{T_3^1} \sqcup 3\mathbf{T_2^1}$	$(0, 6, 1) \sqcup (4, 8, 5) \sqcup (10, 3) \sqcup (9, 7) \sqcup (11, 12)$
$\mathbf{T_3^1} \sqcup 5\mathbf{T_2^1}$	$(0, 6, 1) \sqcup (8, 4) \sqcup (2, 5) \sqcup (10, 3) \sqcup (9, 7) \sqcup (11, 12)$

Chapter 4

$\mathbf{n \equiv 7, 8 \pmod{14}}$

In this chapter, we will use our own constructions based on edge lengths in K_n where $n \equiv 7$ or $8 \pmod{14}$. We first describe our construction in the context of K_{21} and K_{22} in Subsection 4.1.1, and generalize our construction in Subsection 4.1.2. We formalize these ideas in Section 4.2.

4.1 Construction for $n \equiv 7, 8 \pmod{14}$

Because the number of vertices in K_7 and K_8 is less than 9, the minimum number of vertices of a seven edge forest, neither K_7 nor K_8 are decomposable by seven edge forests. So, our base graphs are K_{21} and K_{22} for $n \equiv 7$ and $8 \pmod{14}$, respectively. We show these base graphs in the figure on the following page.

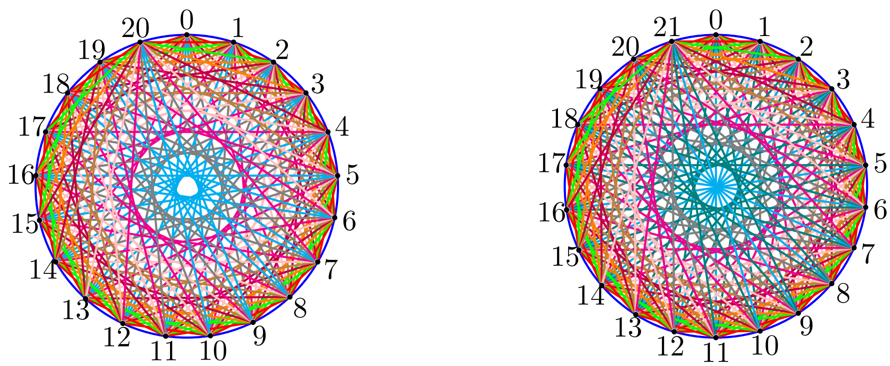

Figure 4.1: K_{21} (left) and K_{22} (right) with edges colored by length.

4.1.1 K_{21} and K_{22}

We have lost some luxuries present in the $n \equiv 0, 1 \pmod{14}$ case. There are two main problems in this case that we need to solve.

(1) In K_{22} there are 11 total maximal length 11 edges but 22 of every other length.

(2) K_{21} and K_{22} have lengths $\{1, \ldots, 10\}$ and $\{1, \ldots, 11\}$, respectively.

(1) is a problem because we want to be able to develop the vertices of our labelings to collect all edges of K_{22} in the same number of steps and (2) is a problem because then we cannot just fit 7 distinct lengths on a single labeling and collect all edges when we develop the vertices by 1. Furthermore, since K_{22} has one more length than K_{21}, we do not have a single labeling strategy that takes care of both cases where $n \equiv 7$ or 8 $\pmod{14}$ at once like σ^{+-}-labelings did for $n \equiv 0$ and 1 $\pmod{14}$. We address these problems in order.

We have a remedy for length 11 edges in K_{22} that is similar to what we did for K_{14}. We take the vertices of K_{22} to be $\mathbb{Z}_{21} \cup \{\infty\}$ and redefine length of edges and development for vertices in K_{22}:

$$\ell(uv) = \begin{cases} \min\{|u-v|, 21-|u-v|\}, & u, v \neq \infty, \\ \infty, & u \text{ or } v = \infty \end{cases} \quad \text{and } v \mapsto \begin{cases} v+1, & v \in \mathbb{Z}_{n-1}, \\ \infty, & v = \infty. \end{cases}$$

Now, we have 21 edges of lengths 1 through 10 as well as ∞, since the ∞ vertex will have all 21 edges of length ∞ to vertices in $\mathbb{Z}_{21}$ adjacent to it. Now we can cyclically generate edges of every length in K_{21} in the same number of steps, as well as all edges of every length in K_{22} in the same number of steps, but we just cannot generate them all on one labeling for either case since the number of edges on our forests does not divide the number of distinct lengths in K_{21} or K_{22}.

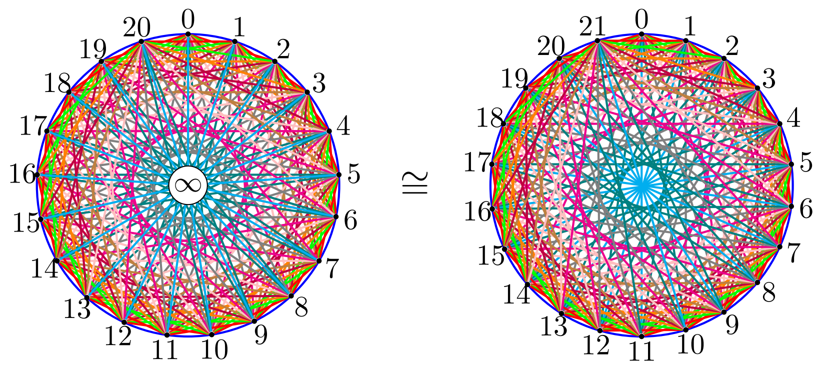

Figure 4.2: $K_{21} \vee K_1$ is isomorphic to K_{22} (right).

Let us begin solving (2) by just looking at K_{21}. If we can generate all 21 distinct edges of lengths $1, 2$ and 3 seperately, then we can generate the remaining 7 lengths with one labeling. But how can we collect 21 edges of length $1, 2,$ and 3 in isomorphic copies? By imposing a new edge length on top of the standard length ℓ, we can can achieve this. We define ℓ_7^+ from $\mathbb{Z}_{21} \cup \{\infty\}$ to $\mathbb{Z}_7$ as follows:

$$\ell_7^+(uv) = \begin{cases} u + v \textbf{ mod } 14, & u, v \neq \infty, \\ v, & u = \infty. \end{cases}$$

Now, every edge has a standard length ℓ and an additive length ℓ_7^+ modulo 7. Previously, we partitioned the edges into sets E_i of edges of length i for each length $i \in \{1, \ldots, 10\}$ via the standard length function ℓ. Now, within each partite set E_i, we have further partitioned the edges into sets $E_{i,j}$ of standard length i and additive length j modulo 7. For example: the edge $(1, 8)$ has length $\ell((1, 8)) = 7$ and $\ell_7^+((1, 8)) = 8 + 1 \textbf{ mod } 7 = 2$, so $(1, 8) \in E_{7,2}$. We show a bundle of labelings of $\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$ that utilizes this new partition on the next page.

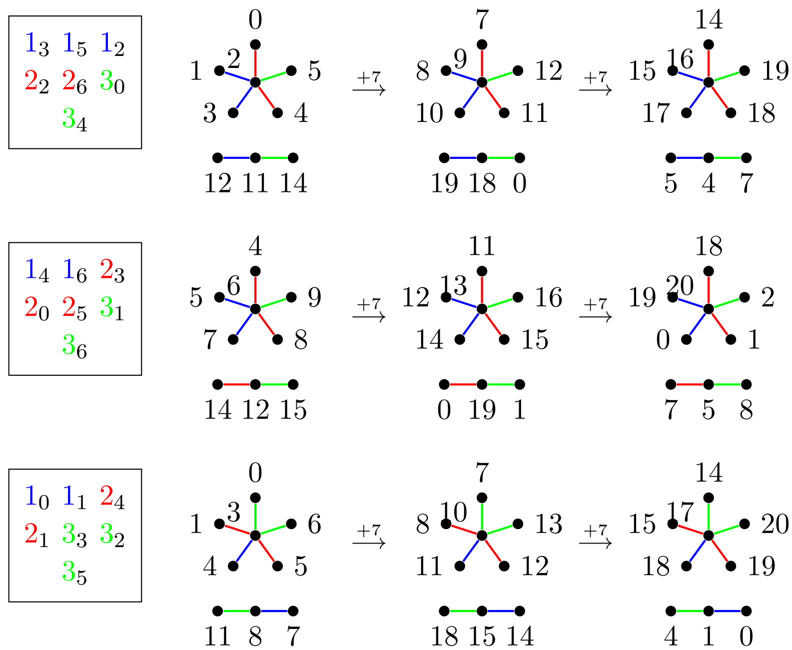

Figure 4.3: Three labelings of $\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$ (left column) that generate all edges of lengths $1, 2,$ and 3 in K_{21} when developed by 7.

Each square in the figure above contains the edge length pairs of the edges in the labelings to its right. Standard lengths ℓ are colored in the same manner as in Figure 4.3 and the new length ℓ_7^+ appears in black as the subscript for each of these lengths. Across all three labelings, there is exactly one representative for all 7 distinct equivalence classes modulo ℓ_7^+ for each length in $\{1, 2, 3\}$. Now, we already know developing the vertices by 7 preserves ℓ, but it also preserves ℓ_7^+ since $u + 7 + v + 7 \textbf{ mod } 7 = u + v \textbf{ mod } 7$. So then developing all labelings by 7 will give us all edges of lengths in $\{1, 2, 3\}$ across 9 isomorphic copies of $\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$.

An equivalent way of understanding development is through *group actions*. Let S be the set of all labelings in Figure 4.3. The cyclic subgroup $\langle 7 \rangle = \{0, 7, 14\} \subseteq \mathbb{Z}_{21}$ acts on S; $\langle 7 \rangle \curvearrowright S$ via developing the vertices by 7. Then, if we call the labelings in column 1: F_1, F_2, F_3 from top to bottom, the set of labelings in Row 1 is simply $\text{Orb}_{\langle 7 \rangle}(F_1)$, Row 2 is $\text{Orb}_{\langle 7 \rangle}(F_2)$ and Row 3 is $\text{Orb}_{\langle 7 \rangle}(F_3)$. We employ the same technique for K_{22}, we just need four labelings F_1, F_2, F_3, F_4 with lengths $1, 2, 3, \infty$.

Now, we have a technique to collect all edges of lengths $\{1, 2, 3\}$ and $\{1, 2, 3, \infty\}$ in K_{21} and K_{22}, respectively, in isomorphic copies of graphs. So, we just need to collect edges of lengths in $\{4, \dots 10\}$. Well, if we just stretch our σ^{+-}-labelings by 3, we will now have lengths $\{4, \dots, 10\}$ instead of $\{1, \dots, 7\}$. So, developing those stretched labelings by 1 will give us the remaining edges of both K_{21} and K_{22} in isomorphic copies of our forests.

4.1.2 Generalizing to K_{14t+7} and K_{14t+8} for $t > 1$

Recall our labeling constructions for collecting edges of lengths $\{1, 2, 3\}$ and $\{1, 2, 3, \infty\}$ in K_{21} and K_{22}, respectively. As long as we only use short edges, we can actually just generate all edges of these lengths for the entire families with the same set of labelings we used for K_{14t+7} and K_{14t+8} where $t > 1$. The only thing that changes at each step $t \mapsto t + 1$ is that more steps of development are needed to collect all the edges, and therefore the number of isomorphic copies.

Equivalently, the orbits of the labelings will simply grow at each step $t \mapsto t + 1$. Let $\langle 7 \rangle_n$ denote the subgroup $\langle 7 \rangle \subset K_n$. Then for each forest and its set of labelings F_1, F_2, F_3 used for K_{21}, $\mathrm{Orb}_{\langle 7 \rangle_{21}}(F_i) \subset \mathrm{Orb}_{\langle 7 \rangle_{14t+7}}(F_i)$ for each $i = 1, 2, 3$ and $t > 1$.

Next, recall how we generalized labelings to K_{14t} and K_{14t+1} in Chapter 3 for edges of subsequent lengths $\bigcup_{0<i<t}\{1+7i, \dots, 7+7i\}$ where $t > 1$. We do the same thing here for K_{14t+7} and K_{14t+8} except for all subsequent edges of lengths in $\bigcup\limits_{0<i<t}\{4+7i, \dots, 10+7i\}$. We simply stretch our labelings (previously σ^{+-}-labelings which were stretched by 3) for both K_{21} and K_{22} that contain edge lengths in $\{4, \dots, 10\}$ by 7 at each step $t \mapsto t+1$ to collect each set of edges with new lengths that come 7 at a time. The nice thing is, we do not need to do anything but this stretching on our one labeling at each step, since we took care of ∞ in K_{22}. Developing all of these by 1 will give us the remaining edges in isomorphic copies of our forests and complete the forest decompositions of K_{14t+7} and K_{14t+8} where $t \geq 1$.

We formalize these constructions in the next section.

4.2 Results for $n \equiv 7, 8 \pmod{14}$

We begin with a formal definition for the set of labelings used to generate edges in K_{14t+7} of lengths in $\{1, 2, 3\}$ in isomorphic copies of a graph G on seven edges for $t \geq 1$. We prove that if a graph G admits such a labeling and a ρ^+-labeling, then it decomposes K_{14t+7} for $t \geq 1$.

Definition 4.2.1. Let G be a graph with 7 edges. A (1-2-3)-*labeling* of $3G$ is an assignment f of the integers $\{0, \ldots, 20\}$ to the vertices of $3G$ such that

(1) $f(u) \neq f(v)$ whenever u and v belong to the same connected component,

and

(2)
$$\bigcup_{uv \in E(3G)} \{(f(u) \textbf{ mod } 7, f(v) \textbf{ mod } 7)\} = \bigcup_{i=0}^{6} \bigcup_{j=1}^{3} \{(i, i+j \textbf{ mod } 7)\}.$$

Notice that the second condition of a (1-2-3)-labeling **demands** that $3G$ contains exactly 7 edges of each length in $\{1, 2, 3\}$. Additionally, the second condition requires that no two edges of the same length have the same end labels modulo 7. A (1-2-3)-labeling of every forest with 7 edges except $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ is given in Table A.1. This exceptional graph does not admit such a labeling, and we deal with it in Chapter 5.

Theorem 4.2.2. *Let G be a bipartite graph with 7 edges. If $3G$ admits a (1-2-3)-labeling and G admits a ρ^+-labeling, then G decomposes K_{14k+7} for every $k \geq 1$.*

Proof. Let $n = 14k+7$ and notice that K_n has $|E(K_n)| = (7k+3)(14k+7)$ edges, which can be partitioned into $14k+7$ edges of each of the lengths in $\{1, 2, \ldots, 7k+3\}$. We will construct the G-decomposition in two steps. First, we use the (1-2-3)-labeling to identify all the edges of lengths $1, 2$, and 3 accounting for $3(2k+1)$ copies of G. Then, we use the ρ^+-labeling to identify edges of the remaining lengths in $7k(2k+1)$ copies of G. In total, the decomposition consists of $|E(K_n)|/7 = (7k+3)(2k+1)$ copies of G.

Let f_1 be a (1-2-3)-labeling of $3G$ and identify this graph as a block B_0. Then develop B_0 by 7 modulo n. Since the order of the development is $\frac{n}{7} = 2k+1$ and there are 7 edges of each of the lengths $1, 2$, and 3 in B_0, we have identified $3(2k+1)$ copies of G containing all $14k+7=n$ edges of each length $1,2$, and 3. Notice (2) of Definition 4.2.1 ensures no edge has been counted more than once in the development.

Let $f_2 : V(G) \to \{0,\dots,14\}$ be a ρ^+-labeling of G with associated vertex partition (A,B). For $i=1,2,\dots,k$, identify blocks $B_i \cong G$ with vertex labels ℓ such that

$$\ell(v) = \begin{cases} f_2(v), & \text{if } v \in A, \\ f_2(v)+3+7(i-1), & \text{if } v \in B. \end{cases}$$

Notice that the i^{th} block contains exactly one edge of each length $7i-3, 7i-2, \dots,$ and $7i+3$. This is because every edge ab has length

$$\ell(b)-\ell(a) = f_2(b)-f_2(a)+3+7(i-1)$$

and $f_2(b)-f_2(a)$ is a length in $\{1,\dots,7\}$. Developing each block B_i by 1 yields $14k+7$ copies of G per block and accounts for $14k+7$ edges of each of the lengths $4,5,\dots,$ and $7k+3$.

Since we have identified

$$3(2k+1)+k(14k+7) = (7k+3)(2k+1)$$

edge-disjoint copies of G, the proof is complete.

$\square$

Next we formalize the set of labelings used to generate edges in K_{14t+8} of lengths in $\{1,2,3,\infty\}$ in isomorphic copies of a graph G on seven edges for $t \geq 1$. Then, we prove that if a graph G admits such a labeling and a ρ^+-labeling, then it decomposes K_{14t+8} for $t \geq 1$.

Definition 4.2.3. Let G be a graph with 7 edges. A *1-rotational* (1-2-3)*-labeling* of $4G$ is an assignment f of $\{0,\dots,20\} \cup \infty$ to the vertices of $4G$ such that

(1) $f(u) \neq f(v)$ whenever u and v belong to the same connected component,

and

(2)

$$\bigcup_{uv \in E(4G)} \{(f(u) \textbf{ mod } 7, f(v) \textbf{ mod } 7)\} = \bigcup_{i=0}^{6} \bigcup_{j=1}^{3} \{(i, i+j \textbf{ mod } 7), (i,\infty)\}.$$

Notice that the second condition of a 1-rotational (1-2-3)-labeling demands that $4G$ contains exactly 7 edges of each length in $\{1, 2, 3, \infty\}$. Additionally, the second condition requires that no two edges of the same length have the same end labels modulo 7. A 1-rotational (1-2-3)-labeling of every forest with 7 edges except $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ is given in Table A.2. This exceptional graph does not admit this labeling either, and we deal with it for this case as well in Chapter 5.

Theorem 4.2.4. *Let G be a bipartite graph with 7 edges. If $4G$ admits a 1-rotational (1-2-3)-labeling and G admits a ρ^+-labeling, then G decomposes K_{14k+8} for every $k \geq 1$.*

Proof. Let $n = 14k+8$ and notice that K_n has $|E(K_n)| = (7k+4)(14k+7)$ edges, which can be partitioned into $14k+7$ edges of each of the lengths in $\{1, 2, \ldots, 7k+3, \infty\}$. We will construct the G-decomposition in two steps. First, we use the 1-rotational (1-2-3)-labeling to identify all the edges of lengths $1, 2, 3$, and ∞ accounting for $4(2k+1)$ copies of G. Then, we use the ρ^+-labeling to identify edges of the remaining lengths in $7k(2k+1)$ copies of G. In total, the decomposition consists of $|E(K_n)|/7 = (7k+4)(2k+1)$ copies of G. Let f_1 be a 1-rotational (1-2-3)-labeling of $4G$ and identify this graph as a block B_0. Then, develop B_0 by 7 modulo $n-1$. Since the order of the development is $\frac{n-1}{7} = 2k+1$ and there are 7 edges of each of the lengths $1, 2, 3$ and ∞ in B_0, we have identified $4(2k+1)$ copies of G containing all $14k+7 = n-1$ edges of each length $1, 2, 3$ and ∞. Notice (2) of Definition 4.2.3 ensures no edge has been counted more than once in the development.

Let $f_2 : V(G) \to \{0, \ldots, 14\}$ be a ρ^+-labeling of G with associated vertex partition (A, B). For $i = 1, 2, \ldots, k$, identify blocks $B_i \cong G$ with vertex labels ℓ such that

$$\ell(v) = \begin{cases} f_2(v), & \text{if } v \in A, \\ f_2(v) + 3 + 7(i-1), & \text{if } v \in B. \end{cases}$$

Notice that the i^{th} block contains exactly one edge of each length $7i-3, 7i-2, \ldots,$ and $7i+3$. This is because every edge ab has length

$$\ell(b) - \ell(a) = f_2(b) - f_2(a) + 3 + 7(i-1)$$

and $f_2(b) - f_2(a)$ is a length in $\{1, \ldots, 7\}$. Developing each block B_i by 1 yields $14k+7$ copies of G per block and accounts for $14k+7$ edges of each of the lengths $4, 5, \ldots,$ and $7k+3$.

Since we have identified

$$4(2k+1)+k(14k+7)=(7k+4)(2k+1)$$

edge-disjoint copies of G, the proof is complete. □

We are now able to state the main theorem of this chapter.

Theorem 4.2.5. *Let F be a forest with 7 edges and $F \ncong \mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$. There exists an F-decomposition of K_n whenever $n \equiv 7$ or $8 \pmod{14}$ and $n \geq 21$.*

Proof. If $n \equiv 7 \pmod{14}$, a (1-2-3)-labeling of $3F$ can be found in Table A.1. On the other hand, if $n \equiv 8 \pmod{14}$, then a 1-rotational (1-2-3)-labeling of $4F$ can be found in Table A.2. In either case, a ρ^+-labeling of F can be found in Table 3.1 (recall that a σ^{+-}-labeling is a ρ^+-labeling). The result now follows from Theorems 4.2.2 and 4.2.4. □

We illustrate how to interpret the tables of labelings and realize the constructions from the last two chapters by building an F-decomposition of K_{35} and K_{36} for the forest graph $\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$.

Example 4.2.6. *Here are excerpts from Tables 3.1,A.1, and A.2 for $\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$*

Labeling Type	Labelings
σ^{+-}	$(4,1,8,5,6,7) \sqcup (10,9,11)$
(1-2-3)	$(0,2,1,3,4,5) \sqcup (12,11,14)$ $(4,6,8,9,5,7) \sqcup (14,12,15)$ $(0,3,1,4,5,6) \sqcup (11,8,7)$
1-rotational (1-2-3)	$(1,2,0,3,4,5) \sqcup (11,8,\infty)$ $(2,\infty,3,4,5,6) \sqcup (12,13,15)$ $(6,7,8,4,5,\infty) \sqcup (11,12,15)$ $(11,10,8,12,13,7) \sqcup (9,6,4)$

Figure 4.4: Labelings for $\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$.

The ρ^+ labelings obtained by stretching the σ^{+-} labeling are bottommost labelings in the following generating presentations and are developed by 1*.*

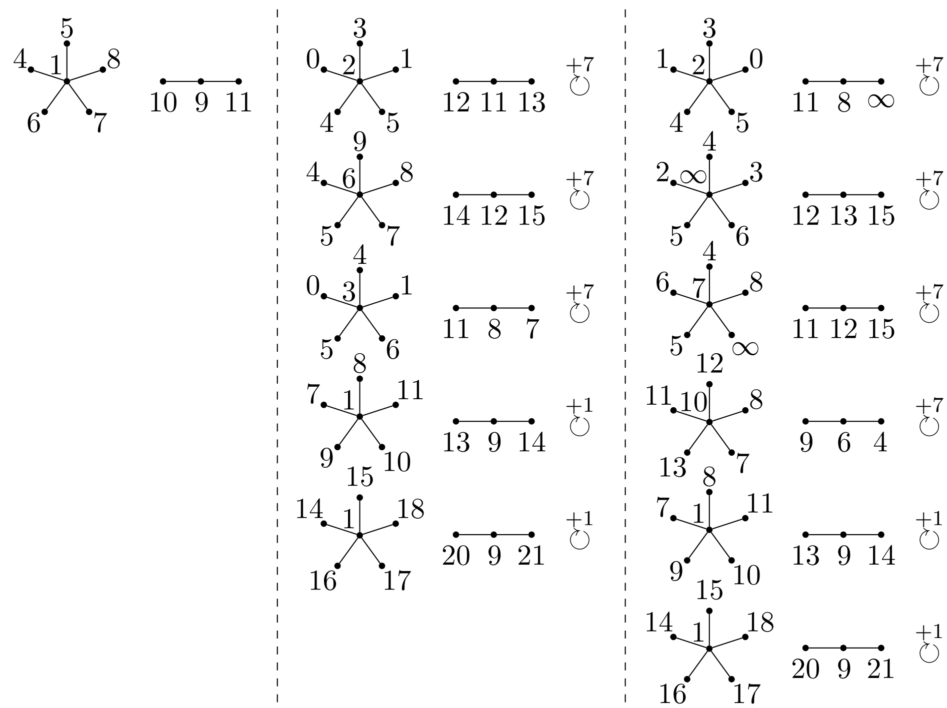

Figure 4.5: A σ^{+-}-labeling of $F \cong \mathbf{T_6^6} \sqcup \mathbf{T_3^1}$ (left) and generating presentations for the F-decomposition of K_n where $n = 35$ (middle) and $n = 36$ (right).

We have now proved that every seven edge forest except $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ decomposes K_n if and only if $n \equiv 0, 1, 7,$ or $8 \pmod{14}$. As stated earlier, we deal with the this exceptional forest $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ in the next chapter.

Chapter 5

$\mathbf{T_7^{11} \sqcup T_2^1}$

To begin we construct K_{14t+7} and K_{14t+8} using joined copies of K_{22}, K_{21}, and K_{14}. Let t be a positive integer. Now join $t-1$ copies of K_{14} with each other and a lone copy of K_{21}. The resulting graph is $K_{14(t-1)+21} \cong K_{14t+7}$. Similarly, K_{14t+8} can be constructed by joining $t-1$ copies of K_{14} with each other and 1 copy of K_{22}.

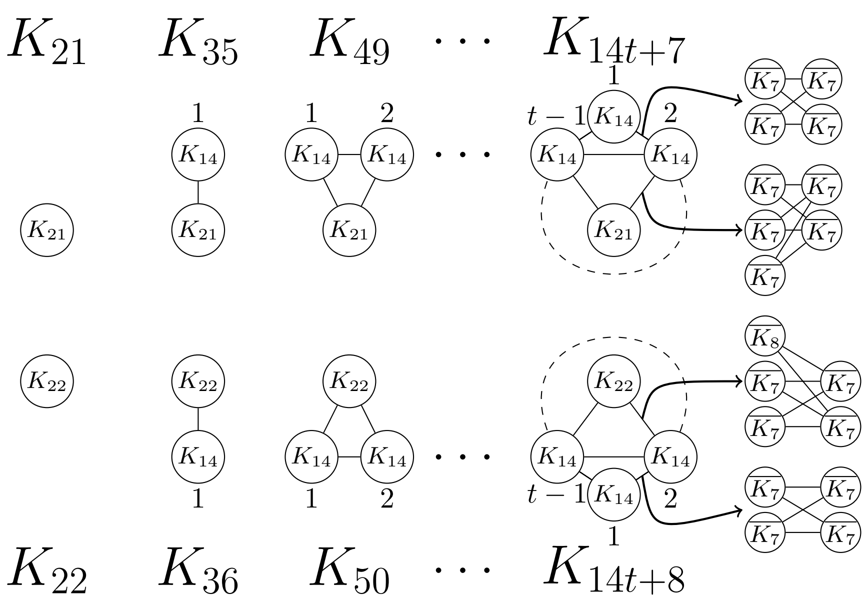

Figure 5.1: A different construction for K_{14t+7} and K_{14t+8}.

Equivalently, We can view K_{14t+7} as K_t whose 'vertices' are K_{14} except one which is K_{21}, and whose edges are the join between them. We will refer to these 'vertices' as nodes. Similarly, we can view K_{14t+8} as K_t whose nodes are K_{14} except one which is K_{22}, and whose edges are the join between them. Notice in Figure 5.1 that all edges in the K_t constructions of these families are then the edges of $K_{14,14}$, $K_{21,14}$, and $K_{22,14}$.

We show that $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ decomposes K_n for $n \equiv 7$ or $8 \pmod{14}$ where $n \geq 21$ by proving that K_{22}, K_{21}, K_{14}, $K_{22,14}$, $K_{21,14}$, and $K_{14,14}$ can each be decomposed by $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ since these six graphs make up the nodes and edges of the K_t representations of K_{14t+7} and K_{14t+8} where $t \geq 1$.

We begin with K_{21} and K_{22}. The proof of the next theorem was obtained by manipulating a $K_{1,7}$-decomposition of K_{21} by Cain in [1]. We 'plucked edges off' of every 7-edge star in the decomposition, put them to the side, and then sent them to 6-edge stars that they were vertex disjoint from. This gave us a $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$-decomposition of K_{21}, which can be found in Table A.3. Then, we just added three 6-edge stars centered at ∞, and three distinct single edge paths from ∞ to the remaining three neighbors of ∞ in K_{22} not covered in the stars. We once again put all the lone paths aside, and sent them to 6-edge stars that they were vertex disjoint from them. This gave us a $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$-decomposition of K_{22} which can be found in Table A.4.

Theorem 5.0.1. $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ *decomposes* K_{21} *and* K_{22}.

Proof. Tables A.3 and A.4 give $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$-decompositions of K_{21} and K_{22}, respectively. □

We did not need to develop anything formally to do this as we only performed this process once for the given 7-edge star decomposition of K_{21}. Initially, we were interested in proving a stronger statement resulting from Cain's work, but found it difficult. Investigating implications of Cain's work [1] from 1974 is something we are interested in for future work.

Next, we address $K_{22,14}$, $K_{21,14}$, and $K_{14,14}$.

Theorem 5.0.2. $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}$ *decomposes* $K_{n,7}$ *for all* $n \geq 2$.

Proof. Consider $K_{n,7}$ where $n \geq 2$. Take the partite set of n vertices to be $\mathbb{Z}_n$ and color them white. Similarly, take the partite set of 7 vertices to be K_7 and color them black. Naturally, we refer to *white-black* vertices uv in $K_{n,7}$ via $(u,v) \in \mathbb{Z}_n \times \mathbb{Z}_7$ and vice versa. Finally, let $E_i = \{(i,0)\} \sqcup (\{i+1\} \times \{1,\ldots,6\})$ and $G_i \subset K_{n,7}$ be the subgraph induced by E_i for each $i \in \mathbb{Z}_n$. Note that $G_i \cong \mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}$ for all $i \in \mathbb{Z}_n$.

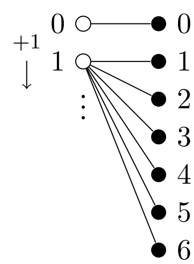

Figure 5.2: G_0 in a generating presentation of the $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}-$decomposition of $K_{n,7}$.

Notice that $E_i \cap E_j = \emptyset$ if $i \neq j$, so, by definition, all distinct G_i's are pairwise edge disjoint. Lastly,

$$\bigcup_{i\in\mathbb{Z}_n} E_i = [\bigcup_{i\in\mathbb{Z}_n} \{(i,0)\}] \sqcup [\bigcup_{i\in\mathbb{Z}_n} (\{i+1\}\times\{1,\ldots,6\})] = [\mathbb{Z}_n\times\{0\}] \sqcup [\mathbb{Z}_n\times\{1,\ldots,6\}] = \mathbb{Z}_n\times\mathbb{Z}_7.$$

So, $G_0 \cup \cdots \cup G_{n-1} = K_{n,7}$ and $\{G_i \mid i \in \mathbb{Z}_n\}$ is a $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}-$decomposition of $K_{n,7}$. Furthermore, it is generated by developing the white vertices of G_0 by 1.

$\square$

Corollary 5.0.3. $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}$ *decomposes* $K_{22,14}$, $K_{21,14}$, *and* $K_{14,14}$.

Proof. $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}$ decomposes $K_{7,7}$ and $K_{8,7}$ by Theorem 5.0.2. $K_{14,14}$ can be expressed as the edge-disjoint union of four copies of $K_{7,7}$; $K_{21,14}$ can be expressed as the edge-disjoint union of six copies of $K_{7,7}$; and $K_{22,14}$ can be expressed as the edge-disjoint union of two copies of $K_{8,7}$ and four copies of $K_{7,7}$. Therefore, $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}$ decomposes them all.

$\square$

Recall that we proved that K_{14} and K_{15} are $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$-decomposable in Chapter 3. So, we are now ready to put everything together to state the main theorem of this chapter, completing the main result of this thesis.

Theorem 5.0.4. $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ *decomposes* K_{14t+7} *and* K_{14t+8} *where* t *is a positive integer.*

Proof. We have that $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ decomposes K_{14} by Theorem 3.2.7; $K_{22,14}$, $K_{21,14}$, and $K_{14,14}$ by Corollary 5.0.3; and lastly K_{22}, K_{21} by Theorem 5.0.1.

Therefore, $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ decomposes the join of $(t-1)$ copies of K_{14} with each other and 1 copy of K_{21}, which is isomorphic to K_{14t+7}. Similarly, $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ decomposes the join of $(t-1)$ copies of K_{14} with each other and 1 copy of K_{22}, which is isomorphic to K_{14t+8}.

□

We have now solved the spectrum problem for every forest on seven edges.

Chapter 6

Additional Results

We present two main additional results produced by work done on this project. (1) Wraparound edge mappings that preserve lengths and (2) Galaxy graph decompositions of bipartite graphs.

6.1 Wraparound edge mappings that preserve lengths

These wraparound mappings only apply to vertices in K_{2mt+m} and $K_{2mt+(m+1)}$, where $m > 1$ is odd. Also, these are meant to be used in labelings that will be developed by m, such as our (1-2-3)-labelings and 1-rotational (1-2-3)-labelings, otherwise one could just map them anywhere. Our approach is a somewhat invasive way to deal with wraparound edges, and it should be noted that it is a barebones sort of framework to begin dealing with wraparound edges in labelings. Note that 'new' short edge mappings will be fixed after the first step from the base graphs. However the 'new' wraparound edges will change at every step.

If uv is a wraparound edge of length l in K_{3m}, its length eventually will not be preserved at some step $t \mapsto t + c$ for $c \geq 1$ in the infinite family of graphs of the form K_{2mt+m}. So the idea is, maybe we can map the wraparound edge to a (1) a short edge of length l at each step $t \mapsto t + 1$ or (2) a wraparound edge of length l at each step $t \mapsto t + 1$ so that lengths will be preserved. The next theorem is something alluded to as an observation to previously, but we prove it anyways.

Theorem 6.1.1. *In K_n, whose vertices we take to be $\mathbb{Z}_n$, uv is a wraparound edge if and only if the absolute difference of its endpoints is greater than the maximal length in K_n; $\lfloor \frac{n}{2} \rfloor < |u-v|$.*

Proof. Let uv be an edge in K_n via ℓ defined previously. If uv is a wraparound edge, then $n - |u-v| < |u-v|$. So, $\frac{n}{2} - \frac{|u-v|}{2} < \frac{|u-v|}{2}$, and therefore $\lfloor \frac{n}{2} \rfloor \leq \frac{n}{2} < |u-v|$. If $\lfloor \frac{n}{2} \rfloor \leq \frac{n}{2} < |u-v|$, note that without loss of generality $u < v$. Then, necessarily $n - |u-v| = n-(v-u) < v-u = |u-v|$, so $n < 2(v-u)$ and $\lfloor \frac{n}{2} \rfloor \leq \frac{n}{2} < (v-u) < |u-v|$.

Thus,

$$\lfloor \frac{n}{2} \rfloor < |u-v| \iff \text{uv is a wraparound edge.}$$

□

We use the notation uv and (u,v) for edges interchangably in the following theorem, which introduces the new wraparound edge mappings.

Theorem 6.1.2. *Let us define the following edge length functions*

$$\ell_n(u,v) = \begin{cases} \min\{|u-v|, n-|u-v|\}, & u,v \in \mathbb{Z}_n, \\ \infty, & u = \infty \ \mathit{or} \ v = \infty, \end{cases}$$

$$\ell_n^+(u,v) = \begin{cases} u+v \,\mathbf{mod}\, n, & u,v \in \mathbb{N}, \\ u \,\mathbf{mod}\, n, & u = \infty, \\ v \,\mathbf{mod}\, n, & v = \infty. \end{cases}$$

Now, let $t > 1$, and let $m > 1$ be odd. Lastly, let $h = 2m(t-1)$. For any wraparound edge (a,b) in K_{21} such that $a < b$, we have

$$\ell_{3m}(a,b) = \ell_{2mt+m}(a-h, b-h) = \ell_{2mt+m}(a+h, b+h), \qquad \text{(short)}$$

$$\ell_{3m}(a,b) = \ell_m^+(a-h, b-h) = \ell_m^+(a+h, b+h). \qquad \text{(wraparound)}$$

That is, these mappings preserve the standard length ℓ and additive length ℓ_m^+ modulo m of $ab \in E(K_{3m})$ in K_{2mt+m} and in $K_{2mt+(m+1)}$ where we take $K_{2mt+(m+1)}$ to be $\mathbb{Z}_{2mt+m} \cup \{\infty\}$.

Proof. Since uv is a wraparound edge where $u < v$, $\ell_{3m}(u, v) = 3m - |u - v| = 3m - (v - u) = 3m + u - v$. Let us simply denote this by $\ell_{ab} = 3m + a - b$. Now, let $k = \lfloor \frac{m}{2} \rfloor = \frac{m-1}{2}$ so the maximal length in K_{3m} is $\lfloor \frac{3m}{2} \rfloor = \lfloor \frac{2m+m}{2} \rfloor = \lfloor \frac{2m}{2} + \frac{m}{2} \rfloor = m + \lfloor \frac{m}{2} \rfloor = m + k$ since $k, m \geq 1$. Suppose $a \geq 2m$. Then, $(2m \leq a < b < 3m)$ implies $(1 \leq b - a < m < m + k)$. But then, $|a - b| < m + k$, the maximal length and so ab is not a wraparound edge, a contradiction. So $a < b < 2m$.

(short): Let $\alpha = a - h, \beta = b \in \mathbb{Z}_{2mt+m}$. Note: $2mt + m - h \equiv 2mt + m - 2m(t - 1) = 2m + m = 3m$. Therefore, $3m + a \equiv (2mt + m - h) + a \equiv a - h \pmod{2mt + m}$. Then, since $1 < t$, we have that $3m + a < 3m + 2m = 2m(2) + m \leq 2mt + m$, and so in fact $\alpha = 3m + a$. Recall that $\beta = b < 3m < 3m + a = \alpha$. Then, we have that $|\alpha - \beta| = \alpha - \beta = (3m + a) - b$. Because $3m < 2mt + m$ for $t > 1$, $\ell_{2mt+m}(\alpha\beta) = \min\{|\alpha - \beta|, 2mt + m - |\alpha - \beta|\} = \min\{3m + a - b, 2mt + m + a - b\} = 3m + a - b = \ell_{ab}$.

(wraparound): Here, let $\alpha = a, \beta = b + h \in \mathbb{Z}_{2mt+m}$. Clearly, $\alpha = a < b + h = \beta$. So, $|\alpha - \beta| = \beta - \alpha = b + h - a$. Recall that ab is a wraparound edge in K_{3m} with $a < b$. Thus, $|a - b| = b - a > m + k$, the maximal length in K_{3m}. Therefore, $|\alpha - \beta| = b - a + h > m + k + h = m + k + 2m(t - 1)$. Now, the maximal length in K_{2mt+m} is $\lfloor \frac{2mt+m}{2} \rfloor = mt + k = m + k + m(t - 1)$. Note that (i) $(2m(t^* - 1))_{t^* \geq 2}$ and (ii) $(m(t^* - 1))_{t^* \geq 2}$ are both arithmetic sequences with increments $2m$ and m, respectively. Both expressions are only equal at $t^* = 1$, and then afterwards (i) increases faster than (ii). So, we see that $2m(t - 1) > m(t - 1)$ for all $t > 1$, and thus, $|\alpha - \beta| = b - a + h > m + k + h = m + k + 2m(t - 1) > m + k + m(t - 1)$, the maximal length in K_{2mt+m} for $t > 1$. So $\alpha\beta$ is a wraparound edge in K_{2mt+m} of length $2mt - m - |\alpha - \beta| = 2mt + m - \beta - \alpha = 2mt + m - (b - a + h) = 2mt + m - 2m(t - 1) + a - b = 3m + a - b = \ell_{ab}$.

$\square$

Note that if we develop the mapped endpoints in (short) and (wraparound) by $c \in \mathbb{N}$, edge lengths ℓ_{2mt+m}, ℓ_m^+ are still preserved. We don't provide any kind of labeling to extend this theorem, but we do provide some 'guardrails' that one can implement in their labeling to avoid certain problems. Besides changing lengths of edges incident to vertices mapped this way in higher order family members, the only thing to worry about is the possibility that an endpoint mapped in a higher order complete graph actually collides with another vertex incident to a short edge (since most labels will presumably not be mapped). This can happen. The following theorem gives way to a theorem that guarantees this will not happen given certain conditions are met.

Theorem 6.1.3. *Let $m > 1$ be odd, $t > 1$, and $h = 2m(t-1)$. Let the vertices of K_{3m} and K_{2mt+m} to be $\mathbb{Z}_{3m}$ and $\mathbb{Z}_{2mt+m}$, respectively, and let a, b be distinct vertices in K_{3m} with $a < b$. Then,*

$$b - a \neq m \text{ or } b \not\equiv a \pmod{m} \Rightarrow a - h \neq b + h \text{ in } \mathbb{Z}_{2mt+m}.$$

Proof. Recall that since $a, b \in \mathbb{Z}_{2m+m}$ and are distinct, $1 \leq b - a < 3m$.

If $b-a \neq m$, suppose $\alpha = \beta = a-h = b+h$. Then $b-a \equiv -2h \equiv -4m(t-1) \pmod{2mt+m}$. Because, $-4m \equiv 2mt + m - 4m \equiv 2mt - 3m \equiv m(2t-3) \pmod{2mt+m}$. So we have that

$$b - a \equiv [m(2t-3)][(t-1)] \pmod{2mt+m}.$$

If $t = 2$, $b - a \equiv m(2(2-3)(2-1)) \equiv m \pmod{2m(2)+m}$ and so for $k \geq 1$, we have $b - a = m$ or $b - a = k(m + 2m(2)) + m = 5mk > 3m$, both contradictions. So $\alpha \neq \beta$.

If $t > 2$, $b - a \equiv [m(2t-3)][(t-1)] \equiv m(2t^2 - 5t + 3) \pmod{2mt+m}$. So then $b-a = m(2t^2-5t+3)$ or $b-a = m(2t^2-5t+3)+k(2mt+m)$ for $t \geq 3$ and $k \geq 1$. Since $t \geq 3$, we have $m(2t^2-5t+3) \geq m(2(3)^2-5(3)+3) = 6m$ and $2mt+m \geq 2m(3)+m = 7m$ because both are strictly increasing for $t > 2$. So, $b - a \geq 6m$ or $b - a \geq 6m + 7mk$ for $t > 2$, both contradictions, since $1 < b - a < 3m$.

Finally, on the other hand if $b \not\equiv a \pmod{m}$, suppose $\alpha = \beta$. Then $a - h \equiv b - h \pmod{2mt+m}$ and $b-a \equiv -2h \equiv -4m(t-1) \pmod{2mt+m}$. Then $b-a \equiv 0 \pmod{m}$ since $m | -4m(t-1)$ and $m | 2mt+m$. But then $b \equiv a \pmod{m}$, a contradiction. So, $\alpha \neq \beta$, and the statement is proved.

□

We are now ready to present the final result in this section.

Corollary 6.1.4. *Let $m > 1$ be odd, $t > 1$ and $h = 2m(t-1)$ and take the vertices of K_{3m} and K_{2mt+m} to be $\mathbb{Z}_{3m}$ and $\mathbb{Z}_{2mt+m}$, respectively. Let a, b be distinct vertices in K_{3m} such that $a < b$ and $a \equiv b \pmod{m}$. Then,*

$$t = 2 \text{ and } |a-b| \neq 2m \iff a \pm h \neq b \text{ and } b \pm h \neq a \text{ in } \mathbb{Z}_{2mt+m}, \tag{i}$$

$$t > 2 \implies a \pm h \neq b \text{ and } b \pm h \neq a \text{ in } \mathbb{Z}_{2mt+m}. \tag{ii}$$

Proof. Let $k = \lfloor \frac{m}{2} \rfloor$. In the proof of Theorem 6.1.2 it is shown that $a - h = 3m + a$, so then, since $0 \leq a < 3m$, $3m \leq 3m + a = a - h$ in K_{2mt+m}. Now, suppose $b \leq m$. Because $a < b \leq m$ and $|a - b| = b - a$, necessarily $1 \leq |a - b| \leq m < m + k$, the maximal length in K_{3m}. But then, by Theorem 6.1.1, ab is not a wraparound edge, a contradiction. So, $b > m$ and $b + h > m + h \geq 3m$.

We prove (i) by contrapositive. For $t = 2$, $h = 2m$ and if $a + h = b$ or $a - h = b$, then $|a - b| = h = 2m$. On the other hand, if $|a - b| = h$, then $a - b = h$ or $a - b = -h$ and $a + h = b$ or $a - h = b$. Thus, the contrapositive holds and (i) is proved.

If $t > 2$, recall that $2mt+m = 3m+2m(t-1) = 3m+h$. Then since $0 \leq u < v < 3m$, we have that $h \leq u+h < v+h < 3m+h = 2mt+m$. Now, $h = 2m(t-1) \geq 2m(3-1) = 4m$ for $t > 2$. So,

$$0 \leq u < v < 3m < 4m \leq u + h < v + h < 3m + h = 2mt + m \text{ for } t > 2.$$

Therefore, $u+h \neq v$ and $v+h \neq u$. Now, since $2mt+m-h = 3m$, $2mt+m+(v-h) \equiv 3m + v \pmod{2mt + m}$ and similarly $2mt + m + (u - h) \equiv 3m + u \pmod{2mt + m}$, we have that $u - h$ and $v - h$ are simply $3m + u$ and $3m + v$, respectively, in $\mathbb{Z}_{2mt+m}$. Also,

$$0 \leq u < v < 3m < 4m \leq 3m + u < 3m + v < 5m \leq 2mt + m \text{ for } t > 2.$$

Therefore, $u - h \neq v$ and $v - h \neq u$ in $\mathbb{Z}_{2mt+m}$, and (ii) is proved.

□

This concludes this section. The next section is a fun result that came from dealing with the exceptional graph in Chapter 5.

6.2 Galaxy Decompositions of Complete Bipartite Graphs

In Chapter 5, we viewed $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$ as a forest, but it is also a galaxy. After finding the labeling in Figure 5.2, it became clear we could generalize this idea to all galaxies. We begin with galaxy decompositions of complete bipartite graphs.

Theorem 6.2.1. *Let N and m be positive integers and consider the complete bipartite $K_{N,m}$ with a finite collection $C = \{G_i \mid i \in \mathbb{Z}_n\}$ of vertex disjoint stars. If $0 < n \leq N$ and $m = \sum_{i\in\mathbb{Z}_n} |E(G_i)|$, then there exists a C-galaxy decomposition of $K_{N,m}$.*

Proof. Take the partite set of N vertices to be $\mathbb{Z}_N$ and color them white. Similarly, take the partite set of m vertices to be $\mathbb{Z}_m$ and color them black. Note that we will be using vertices as group elements. Naturally, we refer to *white-black* vertices uv in $K_{N,m}$ via $(u, v) \in \mathbb{Z}_N \times \mathbb{Z}_m$ and vice versa. Next, let $\mathcal{G} = \bigsqcup_{i\in\mathbb{Z}_n} G_i$, a C-galaxy. Lastly, let $L_0 = \{0, \ldots, |E(G_0)|\}$, $L_i = \{|E(G_{i-1})|, \ldots, |E(G_i)| - 1\}$ for $i \in \mathbb{Z}_n^*$ and $\mathcal{G}_j \subseteq K_{N,m}$ be the subgraph induced by $E_j = \sqcup_{i\in\mathbb{Z}_n}(\{i + j\} \times L_i)$ for each $j \in \mathbb{Z}_N$ and note that these subgraphs are all isomorphic to $\mathcal{G}$.

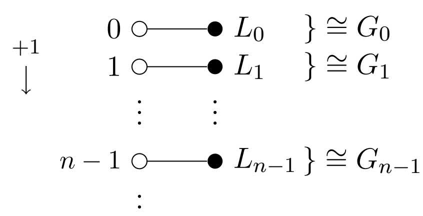

Figure 6.1: $\mathcal{G}_0$ in a generating presentation of the C-galaxy decomposition of $K_{N,m}$.

Consider any edge $(u, v) \in K_{N,m}$. Because $\mathbb{Z}_m = L_0 \sqcup \cdots \sqcup L_{n-1}$, so $v \in L_a$ for exactly one $a \in \mathbb{Z}_n$, $(u, v) \in \{u\} \times L_a = \{a + b\} \times L_i \subseteq \sqcup_{i\in\mathbb{Z}_n}(\{i + b\} \times L_a) = E_b$, where $b = u - a \in \mathbb{Z}_N$. There is only one such $b \in \mathbb{Z}_N$. Therefore, any edge $(u, v) \in K_{N,m}$ belongs to exactly one isomorphic copy $\mathcal{G}_b$ of $\mathcal{G}$ in $\{\mathcal{G}_i \mid i \in \mathbb{Z}_N\}$.

Thus,

$$\{\mathcal{G}_i \mid i \in \mathbb{Z}_N\} \text{ is a } C\text{-galaxy decomposition of } K_{N,m}.$$

$\square$

Next, we extend this idea to complete multipartite graphs.

Corollary 6.2.2. *Let m be a positive integer and $N_0, \ldots, N_{k-1}$ be positive integers that are divisible by m. If $C = \{G_i \mid i \in \mathbb{Z}_n\}$ is a collection of vertex disjoint stars where $0 < n \leq \min\{N_0, \ldots, N_{k-1}\}$ and $m = \sum_{i \in \mathbb{Z}_n} |E(G_i)|$, then there exists a C-galaxy decomposition of the complete k-partite graph $K_{N_0,\ldots,N_{k-1}}$.*

Proof. Let $d_i = \frac{n_i}{m}$ for each $i \in \mathbb{Z}_k$. View $K_{N_1,\ldots,N_k}$ as the union of $\sum\limits_{0 \leq i < j < k} d_i d_j$ copies of $K_{m,m}$. Well, $m = \sum_{i \in \mathbb{Z}_n} |E(G_i)|$ so the maximal number of stars n in C is m, so $n \leq m$ and, by Theorem 6.2.1, the C-galaxy decomposes $K_{m,m}$. Therefore, the C-galaxy decomposes any number of copies of $K_{m,m}$ and so it decomposes $K_{N_1,\ldots,N_k}$. □

This corollary concludes this chapter and the results of this thesis project. We hope to extend this galaxy idea in future work. The next section contains `python` programs made for this project.

Chapter 7

Programming

This project required over 450 labelings of various forests, all of which were found by hand. At around 200 labelings it became clear that due to the sheer number of labelings being done, just probablistically some labelings would (1) have typos (2) have incorrect computations and (3) violate some constraint of the labeling.

All this programming began because we wanted some sort of local program that could display the labelings, so we could check with some level of certainty that they were isomorphic to the forest being worked with. Everything we found that displayed graphs did not allow for dragging vertices and/or interacting with graphs at a high level. So we decided to make our own programs.

There are three groups of programs in a dedicated github repository that we provide links to, for the sake of space. The dependencies required to run these are:

(1) A `python` 3.13 installation,

(2) The `NetworkX` library for `python`,

(3) the `pygame` or `pygame-ce` library for `python`,

(4) the `itertools` library for `python`,

(5) the `z3` library for `python`.

It should be clear when looking at the code what the dependencies are. If you use `pip` in `vscode`, you may simply input the following into your workspace terminal:

```
py -m pip install #enter library name here
# OR
python -m pip install #enter library name here
```

depending on how you install and work with `python`, installing packages may not work this way. `Anaconda` is a popular `python` bundle that likely comes with some of these.

7.1 Tikzgrapher

The file shared here is an earlier version of a program named `tikzgrapher` that was specifically built to display (1-2-3)-labelings and 1-rotational (1-2-3)-labelings. There is a less restrictive version where arguments are optional and customizable (custom edge and vertex labelings, colorings of vertices, no side tab for squares containing lengths). Here is a link to the newer version of `tikzgrapher`:
`https://github.com/tucxy/Programming/tree/main/Python/tikzgrapher`

These are the features ordered earliest to latest in this older build of `tikzgrapher`:

(1) Displays a list of `NetworkX` graphs together on one `pygame` window, starting from top to bottom.

(2) Reduces vertices modulo n and computes the standard edge length for each edge modulo n, and has the subscript as the additive edge length ℓ_7^+.

(3) For each graph in the list given as input: uses a longest path search algorithm and by default displays the longest path of a graph in the center row of a grid of coordinates, then displays vertices coming off of that row.

(4) Has a tab on the left that displays all standard edge lengths ℓ and a chart for the subscript labels of the labelings in order. The window with the tab open looks similar to Figure 4.3 except it doesn't have colored edges.

(5) Allows user to save displayed graphs as a `Tikz` graph in a standalone LaTeX file to a specified path.

We have every single forest labeling in files that import this version `tikzgrapher`. If you simply uncomment below a forest, a `pygame` window will pop out and display the labeling. Here are links to those files.

(1) σ^{+-}-labelings:

`https://github.com/tucxy/Thesis-Programs/blob/main/sigma.py`

(2) (1-2-3)-labelings:

`https://github.com/tucxy/Thesis-Programs/blob/main/7mod14.py`

(3) 1-rotational (1-2-3)-labelings:

`https://github.com/tucxy/Thesis-Programs/blob/main/8mod14.py`

(4) $\mathbf{T}_{\mathbf{7}}^{\mathbf{11}} \sqcup \mathbf{T}_{\mathbf{2}}^{\mathbf{1}}$-decomposition of K_{21} and K_{22}:

`https://github.com/tucxy/Thesis-Programs/blob/main/starpath.py`

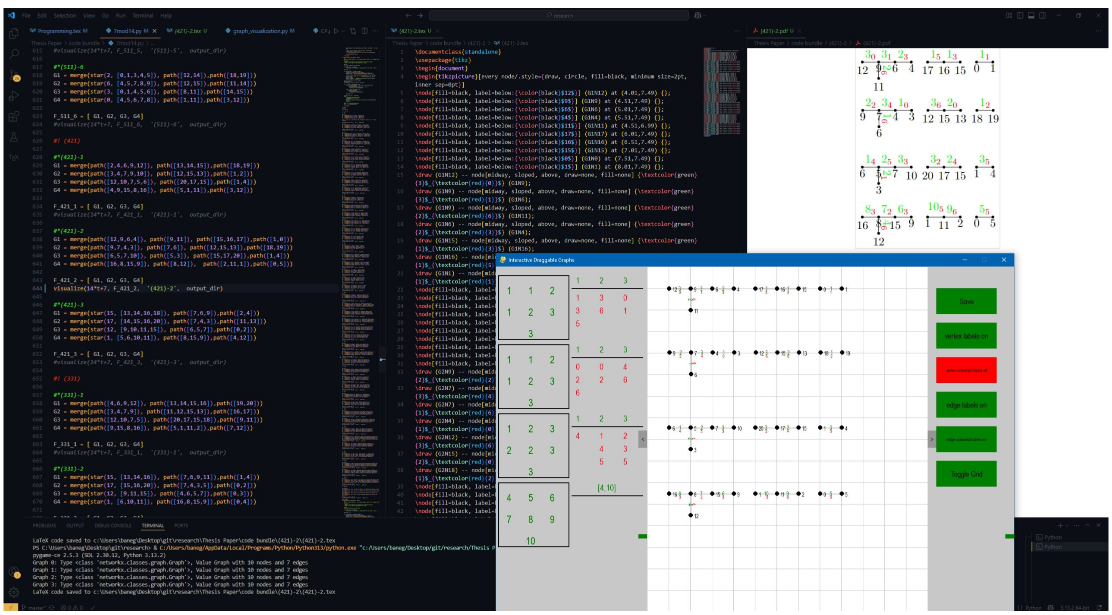

Figure 7.1: A snippet of *tikzgrapher*

Once again, this version is no longer being updated, the following link will take you to the latest version:

`https://github.com/tucxy/Programming/tree/main/Python/tikzgrapher`

7.2 Labeling Solvers

This next program is a bit more ambitious. After all labelings were found (of course) we thought, "Hey what if we didn't have to find these by hand?" Initially, we tried to use a genetic learning algorithm but found that the fitness function was too rigid, and realized that reinforcement learning was not the way to go. We found much more success using constraint programming. Using the `z3` SAT solver, we created (1) a solver that outputs a σ^{+-}-labeling of a graph (2) a graceful labeling of a graph (3) a solver that outputs a more generalized version of the (1-2-3)-labeling for a graph on m edges in K_{2mt+r} where r is an odd idempotent modulo $2m$.

The σ^{+-}-labeling is quite fast, but the other labelings can take up to five minutes or so. In the future, we hope to translate this code to `C++`, to hopefully speed up the process. As of now, it does work and the beefier the processor the better. Here are the links to this project:

(1) Labeling Solvers:
`https://github.com/tucxy/Thesis-Programs/blob/main/CP.py`

(2) Notebook to test the solvers:
`https://github.com/tucxy/Thesis-Programs/blob/main/main.py`

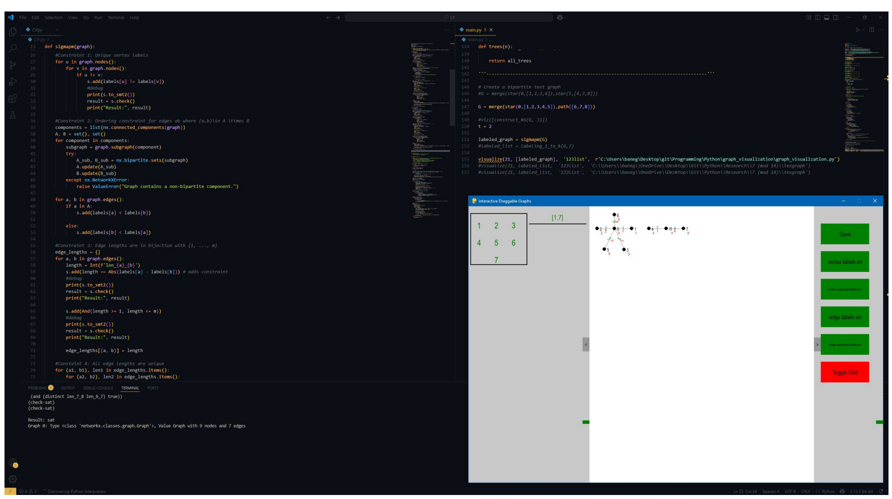

Figure 7.2: A snippet of the σ^{+-}-labeling solver

These are set up so that if you visit this link:
`https://github.com/tucxy/Thesis-Programs/tree/main` and click the code button: the .zip file installed will give a folder when extracted. Make that folder your working directory, and everything should just work.

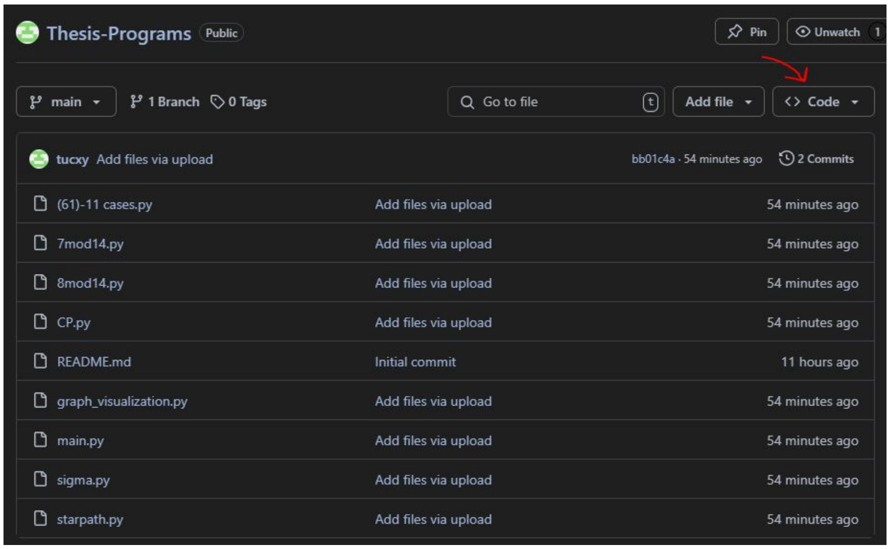

Figure 7.3: click on the code button to download the zip file. Then extract the folder and set it as your working directory.

Feel free to email me with questions: baneg003@outlook.com.

Chapter 8

Conclusion

Using established graph labeling techniques, namely σ^{+-}-labelings and ρ^{+}-labelings, along with our own original constructions and techniques we have proved that every seven edge forest decomposes K_n if and only if $n \equiv 0, 1, 7, \text{ or } 8 \pmod{14}$ and $n \geq 14$. We also proved some results on wraparound edge mappings that preserve lengths across higher order complete graphs in the same families and on galaxy decompositions of complete bipartite graphs. Both of these came out of our work on seven edge forest decompositions of complete graphs. There was also a short chapter on programming that included a new graph visualization software and some labeling solvers.

Of course, a natural continuation of this work would be investigating eight edge forests designs. Additionally, the results on wraparound edge mappings and galaxy graph decompositions are very preliminary and there is a lot open in those areas as well. Specifically, developing a labeling that allows for wraparound edges, and investigating galaxy decompositions of complete graphs. With respect to programming, `tikzgrapher` will continue to be improved but perhaps more exciting and useful are the labeling solvers. Very basic constraint programming algorithms were used for the solvers shared in this paper, so creating more efficient labeling solvers is another area that could be explored and could have a big impact in future research on graph decompositions by eliminating the need to find labelings by hand.

References

[1] P. Cain. Decomposition of complete graphs into stars. *Bull. Austral. Math. Soc.*, 10:23–30, 1974.

[2] S. I. El-Zanati and C. Vanden Eynden. On the cyclic decomposition of complete graphs into bipartite graphs. *Australas. J. Combin.*, 24:209–219, 2001.

[3] S. I. El-Zanati and C. Vanden Eynden. On rosa-type labelings and cyclic graph decompositions. *Math. Slovaca*, 59:1–18, 2009.

[4] B. Freyberg and R. Peters. Decomposition of complete graphs into forests with six edges. *Discuss. Math. Graph Theory*, 34, 2024. in press.

[5] B. Freyberg and N. Tran. Decomposition of complete graphs into bipartite unicyclic graphs with eight edges. *J. Combin. Math. Combin. Comput.*, 114:133–142, 2020.

[6] D. Froncek and M. Kubesa. Decomposition of complete graphs into connected unicyclic bipartite graphs with seven edges. *Bull. Inst. Combin. Appl.*, 93:52–80, 2021.

[7] A. Rosa. On certain valuations of the vertices of a graph. In *Theory of Graphs (Intl. Symp. Rome 1966)*, pages 349–355. Gordon and Breach, Dunod, Paris, 1967.

[8] T. P. Kirkman. On a problem in combinations. *The Lady's and Gentleman's Diary*, pages 48–50, 1850.

[9] T. P. Kirkman. Answer to query vi. *The Lady's and Gentleman's Diary*, page 48, 1851.

[10] D. de Werra. On some combinatorial problems arising in scheduling. *INFOR*, 8:165–175, 1970.

Appendix A

Labelings

A.1 (1-2-3)-labelings

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_7^1} \sqcup \mathbf{T_2^1}$	$(0, 1, 2, 4, 6, 9, 12) \sqcup (13, 14)$ $(3, 4, 7, 9, 10, 13, 15) \sqcup (8, 5)$ $(8, 11, 12, 10, 7, 5, 6) \sqcup (1, 3)$ $(0, 4, 9, 15, 8, 16, 7) \sqcup (1, 11)$
$\mathbf{T_7^3} \sqcup \mathbf{T_2^1}$	$(12, 9, 6, 4, 2, 1, 7) \sqcup (14, 15)$ $(15, 13, 10, 9, 7, 4, 11) \sqcup (8, 5)$ $(8, 11, 12, 10, 7, 5, 13) \sqcup (1, 3)$ $(16, 8, 15, 9, 4, 0, 6) \sqcup (1, 11)$
$\mathbf{T_7^2} \sqcup \mathbf{T_2^1}$	$(0, 1, 2, 4, 6, 9, 3) \sqcup (16, 19)$ $(15, 13, 10, 9, 7, 4, 14) \sqcup (17, 18)$ $(6, 5, 7, 10, 12, 11, 8) \sqcup (18, 15)$ $(7, 16, 8, 15, 9, 4, 12) \sqcup (1, 11)$

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_7^4} \sqcup \mathbf{T_2^1}$	$(8, 6, 4, 2, 1, 9, 7) \sqcup (14, 15)$
	$(8, 10, 9, 7, 4, 11, 13) \sqcup (12, 15)$
	$(9, 12, 10, 7, 5, 11, 13) \sqcup (1, 4)$
	$(7, 15, 9, 4, 0, 8, 6) \sqcup (1, 11)$
$\mathbf{T_7^5} \sqcup \mathbf{T_2^1}$	$(2, 4, 6, 9, 12, 8, 7) \sqcup (11, 14)$
	$(0, 2, 3, 6, 5, 1, 4) \sqcup (8, 7)$
	$(0, 3, 5, 4, 1, 8, 7) \sqcup (16, 15)$
	$(4, 9, 15, 8, 12, 6, 7) \sqcup (1, 11)$
$\mathbf{T_7^8} \sqcup \mathbf{T_2^1}$	$(1, 2, 4, 6, 8, 5, 9) \sqcup (12, 15)$
	$(4, 7, 9, 10, 11, 8, 13) \sqcup (1, 3)$
	$(5, 7, 10, 12, 11, 6, 13) \sqcup (1, 4)$
	$(0, 4, 9, 15, 8, 12, 6) \sqcup (1, 11)$
$\mathbf{T_7^9} \sqcup \mathbf{T_2^1}$	$(8, 6, 4, 2, 5, 9, 7) \sqcup (12, 14)$
	$(1, 3, 2, 0, 5, 4, 6) \sqcup (10, 12)$
	$(9, 8, 7, 10, 4, 11, 5) \sqcup (12, 13)$
	$(7, 15, 9, 4, 13, 8, 6) \sqcup (1, 11)$
$\mathbf{T_7^{10}} \sqcup \mathbf{T_2^1}$	$(7, 6, 4, 2, 8, 9, 5) \sqcup (12, 14)$
	$(2, 3, 4, 7, 0, 5, 6) \sqcup (9, 12)$
	$(7, 8, 5, 4, 9, 10, 11) \sqcup (0, 2)$
	$(6, 15, 9, 4, 8, 11, 7) \sqcup (2, 12)$
$\mathbf{T_7^6} \sqcup \mathbf{T_2^1}$	$(2, 4, 6, 8, 7, 9, 12) \sqcup (13, 14)$
	$(0, 2, 3, 4, 7, 6, 5) \sqcup (8, 10)$
	$(0, 3, 5, 8, 9, 4, 1) \sqcup (12, 14)$
	$(4, 9, 15, 8, 12, 7, 16) \sqcup (1, 11)$
$\mathbf{T_7^7} \sqcup \mathbf{T_2^1}$	$(2, 4, 6, 9, 12, 1, 8) \sqcup (14, 15)$
	$(5, 6, 3, 2, 0, 7, 4) \sqcup (8, 9)$
	$(0, 3, 5, 4, 7, 1, 8) \sqcup (12, 14)$
	$(4, 9, 15, 8, 12, 18, 7) \sqcup (1, 11)$

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_6^1} \sqcup \mathbf{T_3^1}$	$(1,2,4,6,9,12) \sqcup (13,14,15)$
	$(3,4,7,9,10,13) \sqcup (5,8,6)$
	$(11,12,10,7,5,6) \sqcup (3,1,4)$
	$(0,4,9,15,8,16) \sqcup (1,11,2)$
$\mathbf{T_6^2} \sqcup \mathbf{T_3^1}$	$(1,2,4,6,9,5) \sqcup (13,14,15)$
	$(13,10,9,7,4,11) \sqcup (5,8,6)$
	$(11,12,10,7,5,13) \sqcup (3,1,4)$
	$(0,4,9,15,8,12) \sqcup (1,11,2)$
$\mathbf{T_6^3} \sqcup \mathbf{T_3^1}$	$(0,1,2,4,6,5) \sqcup (16,13,14)$
	$(8,6,3,2,0,4) \sqcup (14,12,15)$
	$(7,4,5,3,0,6) \sqcup (10,8,11)$
	$(7,0,4,9,15,12) \sqcup (1,11,2)$
$\mathbf{T_6^4} \sqcup \mathbf{T_3^1}$	$(1,2,5,4,6,7) \sqcup (16,14,13)$
	$(8,6,9,3,2,4) \sqcup (14,12,15)$
	$(4,5,6,3,0,1) \sqcup (11,8,7)$
	$(7,0,6,4,9,12) \sqcup (1,11,2)$
$\mathbf{T_6^5} \sqcup \mathbf{T_3^1}$	$(0,2,4,7,1,5) \sqcup (12,11,13)$
	$(7,6,3,2,8,9) \sqcup (14,12,15)$
	$(4,3,5,6,0,1) \sqcup (11,8,7)$
	$(8,0,4,9,6,7) \sqcup (1,11,2)$
$\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$	$(0,2,1,3,4,5) \sqcup (12,11,14)$
	$(4,6,8,9,5,7) \sqcup (14,12,15)$
	$(0,3,1,4,5,6) \sqcup (11,8,7)$
	$(4,0,8,5,6,7) \sqcup (1,11,2)$
$\mathbf{T_5^1} \sqcup \mathbf{T_4^1}$	$(2,4,6,9,12) \sqcup (16,15,14,13)$
	$(3,4,7,9,10) \sqcup (11,12,15,13)$
	$(12,10,7,5,6) \sqcup (18,15,17,20)$
	$(4,9,15,8,16) \sqcup (2,11,1,5)$

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_5^2} \sqcup \mathbf{T_4^1}$	$(12, 9, 6, 4, 11) \sqcup (17, 16, 15, 14)$ $(9, 7, 4, 3, 6) \sqcup (11, 12, 15, 13)$ $(6, 5, 7, 10, 3) \sqcup (18, 15, 17, 20)$ $(16, 8, 15, 9, 12) \sqcup (2, 11, 1, 6)$
$\mathbf{T_5^2} \sqcup \mathbf{T_4^2}$	$(4, 6, 9, 11, 8) \sqcup (16, 15, 18, 14)$ $(9, 7, 4, 3, 6) \sqcup (16, 17, 20, 15)$ $(6, 5, 7, 10, 3) \sqcup (9, 12, 11, 15)$ $(16, 8, 15, 9, 12) \sqcup (10, 1, 11, 6)$
$\mathbf{T_5^3} \sqcup \mathbf{T_4^1}$	$(13, 15, 16, 18, 14) \sqcup (11, 9, 6, 7)$ $(14, 17, 16, 20, 15) \sqcup (9, 7, 4, 3)$ $(9, 12, 10, 11, 15) \sqcup (4, 6, 5, 7)$ $(5, 1, 10, 11, 6) \sqcup (16, 8, 15, 9)$
$\mathbf{T_5^1} \sqcup \mathbf{T_4^2}$	$(7, 6, 9, 11, 8) \sqcup (16, 15, 13, 14)$ $(9, 7, 4, 3, 5) \sqcup (16, 17, 20, 15)$ $(4, 6, 5, 7, 10) \sqcup (9, 12, 11, 15)$ $(16, 8, 15, 9, 5) \sqcup (10, 1, 11, 6)$
$\mathbf{T_5^3} \sqcup \mathbf{T_4^2}$	$(13, 15, 16, 18, 14) \sqcup (11, 9, 12, 6)$ $(18, 17, 16, 20, 15) \sqcup (9, 7, 10, 4)$ $(10, 12, 11, 14, 15) \sqcup (4, 6, 5, 7)$ $(5, 1, 10, 11, 6) \sqcup (16, 8, 14, 15)$
$\mathbf{T_6^1} \sqcup 2\mathbf{T_2^1}$	$(1, 2, 4, 6, 9, 12) \sqcup (13, 14) \sqcup (8, 7)$ $(3, 4, 7, 9, 10, 13) \sqcup (8, 6) \sqcup (12, 15)$ $(11, 12, 10, 7, 5, 6) \sqcup (1, 4) \sqcup (17, 15)$ $(0, 4, 9, 15, 8, 16) \sqcup (1, 11) \sqcup (3, 12)$
$\mathbf{T_6^2} \sqcup 2\mathbf{T_2^1}$	$(1, 2, 4, 6, 9, 5) \sqcup (13, 14) \sqcup (8, 7)$ $(13, 10, 9, 7, 4, 11) \sqcup (8, 6) \sqcup (12, 15)$ $(11, 12, 10, 7, 5, 13) \sqcup (1, 4) \sqcup (17, 15)$ $(0, 4, 9, 15, 8, 12) \sqcup (1, 11) \sqcup (5, 14)$

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_6^3} \sqcup 2\mathbf{T_2^1}$	$(0,1,2,4,7,5) \sqcup (9,6) \sqcup (8,10)$ $(8,6,3,2,0,4) \sqcup (5,7) \sqcup (12,13)$ $(6,4,5,3,0,8) \sqcup (13,14) \sqcup (18,15)$ $(7,0,4,9,15,12) \sqcup (1,11) \sqcup (5,14)$
$\mathbf{T_6^4} \sqcup 2\mathbf{T_2^1}$	$(1,2,5,4,6,7) \sqcup (13,14) \sqcup (12,15)$ $(8,6,9,3,2,4) \sqcup (12,14) \sqcup (18,15)$ $(4,5,6,3,0,1) \sqcup (8,7) \sqcup (16,14)$ $(7,0,6,4,9,12) \sqcup (1,11) \sqcup (5,14)$
$\mathbf{T_6^5} \sqcup 2\mathbf{T_2^1}$	$(0,2,4,7,1,5) \sqcup (11,13) \sqcup (12,15)$ $(7,6,3,2,8,9) \sqcup (11,12) \sqcup (1,4)$ $(4,3,5,6,0,1) \sqcup (8,7) \sqcup (12,14)$ $(8,0,4,9,6,7) \sqcup (1,11) \sqcup (5,14)$
$\mathbf{T_6^6} \sqcup 2\mathbf{T_2^1}$	$(0,2,1,3,4,5) \sqcup (12,14) \sqcup (18,19)$ $(4,6,8,9,5,7) \sqcup (12,15) \sqcup (11,14)$ $(0,3,1,4,5,6) \sqcup (8,11) \sqcup (14,15)$ $(4,0,8,5,6,7) \sqcup (1,11) \sqcup (3,12)$
$\mathbf{T_5^1} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(2,4,6,9,12) \sqcup (13,14,15) \sqcup (18,19)$ $(3,4,7,9,10) \sqcup (12,15,13) \sqcup (1,2)$ $(12,10,7,5,6) \sqcup (20,17,15) \sqcup (1,4)$ $(4,9,15,8,16) \sqcup (11,1,5) \sqcup (3,12)$
$\mathbf{T_5^2} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(12,9,6,4,11) \sqcup (17,16,15) \sqcup (0,1)$ $(9,7,4,3,6) \sqcup (12,15,13) \sqcup (18,19)$ $(6,5,7,10,3) \sqcup (20,17,15) \sqcup (1,4)$ $(16,8,15,9,12) \sqcup (1,11,2) \sqcup (0,5)$
$\mathbf{T_5^3} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(13,15,16,18,14) \sqcup (9,6,7) \sqcup (2,4)$ $(14,17,16,20,15) \sqcup (3,4,7) \sqcup (11,13)$ $(9,12,10,11,15) \sqcup (6,5,7) \sqcup (0,2)$ $(5,1,10,11,6) \sqcup (8,15,9) \sqcup (4,12)$

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$2\mathbf{T_4^1} \sqcup \mathbf{T_2^1}$	$(4,6,9,12) \sqcup (16,15,14,13) \sqcup (19,20)$ $(9,7,4,3) \sqcup (11,12,15,13) \sqcup (16,17)$ $(12,10,7,5) \sqcup (18,15,17,20) \sqcup (9,11)$ $(9,15,8,16) \sqcup (2,11,1,5) \sqcup (12,7)$
$\mathbf{T_4^1} \sqcup \mathbf{T_4^2} \sqcup \mathbf{T_2^1}$	$(11,9,6,7) \sqcup (16,15,13,14) \sqcup (1,4)$ $(5,3,4,7) \sqcup (16,17,20,15) \sqcup (0,2)$ $(4,6,5,7) \sqcup (9,12,11,15) \sqcup (0,3)$ $(16,8,15,9) \sqcup (10,1,11,6) \sqcup (0,4)$
$2\mathbf{T_4^2} \sqcup \mathbf{T_2^1}$	$(18,15,13,14) \sqcup (11,9,12,6) \sqcup (1,2)$ $(18,17,20,15) \sqcup (9,7,10,4) \sqcup (2,3)$ $(11,12,14,15) \sqcup (4,6,5,7) \sqcup (17,19)$ $(11,1,5,6) \sqcup (16,8,14,15) \sqcup (0,9)$
$\mathbf{T_4^1} \sqcup 2\mathbf{T_3^1}$	$(16,15,14,13) \sqcup (0,3,5) \sqcup (12,9,6)$ $(11,12,15,13) \sqcup (10,9,7) \sqcup (16,18,20)$ $(18,15,17,20) \sqcup (10,11,14) \sqcup (6,5,7)$ $(2,12,3,11) \sqcup (8,1,7) \sqcup (4,0,5)$
$\mathbf{T_4^2} \sqcup 2\mathbf{T_3^1}$	$(11,9,12,6) \sqcup (18,15,13) \sqcup (0,1,2)$ $(9,7,10,4) \sqcup (18,17,20) \sqcup (1,3,2)$ $(11,12,14,15) \sqcup (4,6,7) \sqcup (17,19,20)$ $(16,8,14,15) \sqcup (11,1,6) \sqcup (9,0,4)$
$\mathbf{T_4^1} \sqcup \mathbf{T_3^1} \sqcup 2\mathbf{T_2^1}$	$(8,6,9,11) \sqcup (0,1,2) \sqcup (16,19) \sqcup (18,15)$ $(8,10,7,9) \sqcup (18,17,20) \sqcup (11,14) \sqcup (2,3)$ $(13,11,12,14) \sqcup (17,19,20) \sqcup (6,7) \sqcup (8,5)$ $(0,5,1,7) \sqcup (3,10,2) \sqcup (4,13) \sqcup (16,6)$
$\mathbf{T_4^2} \sqcup \mathbf{T_3^1} \sqcup 2\mathbf{T_2^1}$	$(11,9,12,6) \sqcup (0,1,2) \sqcup (18,15) \sqcup (13,14)$ $(9,7,10,4) \sqcup (18,17,20) \sqcup (11,13) \sqcup (2,3)$ $(11,12,14,15) \sqcup (17,19,20) \sqcup (8,6) \sqcup (1,3)$ $(4,0,5,6) \sqcup (8,1,9) \sqcup (3,12) \sqcup (17,7)$

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_5^1} \sqcup 3\mathbf{T_2^1}$	$(2,4,6,9,12) \sqcup (13,14) \sqcup (18,19) \sqcup (0,1)$ $(3,4,7,9,10) \sqcup (13,15) \sqcup (1,2) \sqcup (8,5)$ $(6,5,7,10,12) \sqcup (17,20) \sqcup (8,11) \sqcup (1,3)$ $(4,9,15,8,16) \sqcup (1,11) \sqcup (3,12) \sqcup (2,6)$
$\mathbf{T_5^2} \sqcup 3\mathbf{T_2^1}$	$(11,9,6,4,12) \sqcup (16,15) \sqcup (8,10) \sqcup (2,3)$ $(6,7,4,3,9) \sqcup (13,15) \sqcup (18,19) \sqcup (8,5)$ $(3,5,7,10,6) \sqcup (17,20) \sqcup (8,11) \sqcup (0,1)$ $(12,8,15,9,16) \sqcup (2,11) \sqcup (0,5) \sqcup (3,13)$
$\mathbf{T_5^3} \sqcup 3\mathbf{T_2^1}$	$(13,15,16,18,14) \sqcup (9,6) \sqcup (2,4) \sqcup (5,7)$ $(14,17,16,20,15) \sqcup (4,7) \sqcup (11,13) \sqcup (5,6)$ $(9,12,10,11,15) \sqcup (6,7) \sqcup (0,2) \sqcup (3,4)$ $(5,1,10,11,6) \sqcup (9,15) \sqcup (4,12) \sqcup (0,7)$
$3\mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(18,15,13) \sqcup (11,9,6) \sqcup (0,1,2) \sqcup (16,19)$ $(18,17,20) \sqcup (9,7,10) \sqcup (1,3,2) \sqcup (11,14)$ $(11,12,14) \sqcup (4,6,7) \sqcup (17,19,20) \sqcup (8,5)$ $(11,1,6) \sqcup (16,8,14) \sqcup (9,0,4) \sqcup (10,3)$
$\mathbf{T_4^1} \sqcup 4\mathbf{T_2^1}$	$(9,6,4,2) \sqcup (13,14) \sqcup (18,19) \sqcup (0,1) \sqcup (10,12)$ $(9,7,4,3) \sqcup (13,15) \sqcup (1,2) \sqcup (8,5) \sqcup (16,17)$ $(10,7,5,6) \sqcup (17,20) \sqcup (8,11) \sqcup (1,3) \sqcup (9,12)$ $(9,15,8,16) \sqcup (1,11) \sqcup (3,12) \sqcup (2,6) \sqcup (0,5)$
$\mathbf{T_4^2} \sqcup 4\mathbf{T_2^1}$	$(16,15,18,13) \sqcup (9,6) \sqcup (2,4) \sqcup (5,7) \sqcup (0,1)$ $(16,17,20,14) \sqcup (4,7) \sqcup (11,13) \sqcup (5,6) \sqcup (1,3)$ $(9,12,10,11) \sqcup (6,7) \sqcup (0,2) \sqcup (3,4) \sqcup (8,5)$ $(10,1,11,5) \sqcup (9,15) \sqcup (4,12) \sqcup (0,7) \sqcup (8,3)$
$2\mathbf{T_3^1} \sqcup 3\mathbf{T_2^1}$	$(11,9,6) \sqcup (0,1,2) \sqcup (18,15) \sqcup (16,19) \sqcup (17,20)$ $(9,7,10) \sqcup (1,3,2) \sqcup (17,18) \sqcup (11,14) \sqcup (8,5)$ $(11,12,14) \sqcup (4,6,7) \sqcup (19,20) \sqcup (13,15) \sqcup (3,5)$ $(11,1,6) \sqcup (16,8,14) \sqcup (0,9) \sqcup (10,3) \sqcup (17,13)$

Table A.1: (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_3^1} \sqcup 5\mathbf{T_2^1}$	$(0,1,2) \sqcup (18,15) \sqcup (9,11) \sqcup (16,19) \sqcup (5,6) \sqcup (10,7)$
	$(1,3,2) \sqcup (17,18) \sqcup (9,7) \sqcup (11,14) \sqcup (8,5) \sqcup (16,13)$
	$(4,6,7) \sqcup (12,14) \sqcup (3,5) \sqcup (13,15) \sqcup (17,20) \sqcup (18,19)$
	$(16,8,14) \sqcup (1,11) \sqcup (0,9) \sqcup (10,3) \sqcup (17,13) \sqcup (2,7)$

A.2 1-rotational (1-2-3)-labelings

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_7^1} \sqcup \mathbf{T_2^1}$	$(0,1,\infty,2,4,5,3) \sqcup (12,15)$
	$(0,2,5,\infty,6,4,1) \sqcup (10,11)$
	$(5,7,\infty,3,6,9,10) \sqcup (13,14)$
	$(\infty,4,7,10,8,6,5) \sqcup (16,15)$
	$(0,4,9,15,8,16,7) \sqcup (1,11)$
$\mathbf{T_7^3} \sqcup \mathbf{T_2^1}$	$(3,5,4,2,\infty,8,1) \sqcup (12,15)$
	$(4,6,\infty,5,2,0,18) \sqcup (10,11)$
	$(10,9,6,3,\infty,0,7) \sqcup (12,14)$
	$(5,6,8,10,7,4,9) \sqcup (0,1)$
	$(16,8,15,9,4,0,6) \sqcup (1,11)$
$\mathbf{T_7^2} \sqcup \mathbf{T_2^1}$	$(3,5,4,2,\infty,1,6) \sqcup (9,10)$
	$(0,2,5,\infty,6,4,1) \sqcup (10,11)$
	$(5,7,\infty,3,6,9,8) \sqcup (13,14)$
	$(\infty,4,7,10,8,6,1) \sqcup (12,15)$
	$(7,16,8,15,9,4,12) \sqcup (1,11)$
$\mathbf{T_7^4} \sqcup \mathbf{T_2^1}$	$(1,2,4,5,8,0,\infty) \sqcup (11,13)$
	$(4,\infty,5,2,3,8,6) \sqcup (16,13)$
	$(6,7,\infty,10,13,8,5) \sqcup (19,20)$
	$(11,10,7,4,1,8,12) \sqcup (13,15)$
	$(7,15,9,4,0,8,6) \sqcup (1,11)$
$\mathbf{T_7^5} \sqcup \mathbf{T_2^1}$	$(5,4,2,3,6,0,1) \sqcup (9,\infty)$
	$(2,5,\infty,6,4,8,11) \sqcup (16,13)$
	$(10,\infty,7,8,11,5,6) \sqcup (12,13)$
	$(4,7,10,8,5,11,12) \sqcup (13,15)$
	$(4,9,15,8,12,6,7) \sqcup (1,11)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_7^8} \sqcup \mathbf{T_2^1}$	$(8,5,4,2,0,6,\infty) \sqcup (11,13)$ $(3,2,5,\infty,8,1,6) \sqcup (16,13)$ $(5,7,\infty,3,4,8,6) \sqcup (13,14)$ $(\infty,4,7,10,8,1,12) \sqcup (13,15)$ $(0,4,9,15,8,12,6) \sqcup (1,11)$
$\mathbf{T_7^9} \sqcup \mathbf{T_2^1}$	$(1,2,4,5,7,0,3) \sqcup (8,11)$ $(11,\infty,6,4,5,8,12) \sqcup (10,13)$ $(6,7,\infty,10,2,8,5) \sqcup (9,12)$ $(11,10,8,5,6,12,7) \sqcup (16,13)$ $(7,15,9,4,13,8,6) \sqcup (1,11)$
$\mathbf{T_7^{10}} \sqcup \mathbf{T_2^1}$	$(1,2,4,6,0,3,5) \sqcup (8,11)$ $(11,\infty,6,5,8,2,12) \sqcup (13,15)$ $(6,7,\infty,10,8,4,5) \sqcup (11,12)$ $(11,10,8,5,12,13,7) \sqcup (9,6)$ $(6,15,9,4,8,11,7) \sqcup (2,12)$
$\mathbf{T_7^6} \sqcup \mathbf{T_2^1}$	$(5,4,2,0,1,3,6) \sqcup (9,\infty)$ $(4,6,\infty,1,2,12,13) \sqcup (8,11)$ $(10,\infty,7,5,3,6,9) \sqcup (13,15)$ $(5,8,10,11,\infty,7,4) \sqcup (9,12)$ $(4,9,15,8,12,7,16) \sqcup (1,11)$
$\mathbf{T_7^7} \sqcup \mathbf{T_2^1}$	$(5,4,2,3,6,\infty,0) \sqcup (8,7)$ $(13,12,\infty,6,4,10,1) \sqcup (8,11)$ $(10,\infty,7,6,9,2,5) \sqcup (13,15)$ $(5,8,10,7,4,9,11) \sqcup (16,19)$ $(4,9,15,8,12,18,7) \sqcup (1,11)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_6^1} \sqcup \mathbf{T_3^1}$	$(3,5,4,2,\infty,1) \sqcup (13,12,15)$
	$(0,2,5,\infty,6,4) \sqcup (8,11,10)$
	$(5,7,\infty,3,6,9) \sqcup (13,14,15)$
	$(\infty,4,7,10,8,6) \sqcup (17,16,15)$
	$(0,4,9,15,8,16) \sqcup (1,11,2)$
$\mathbf{T_6^2} \sqcup \mathbf{T_3^1}$	$(\infty,2,4,5,8,0) \sqcup (11,13,12)$
	$(6,\infty,5,2,3,8) \sqcup (13,16,15)$
	$(6,3,\infty,7,5,4) \sqcup (13,14,15)$
	$(8,10,7,4,\infty,12) \sqcup (18,15,13)$
	$(0,4,9,15,8,12) \sqcup (1,11,2)$
$\mathbf{T_6^3} \sqcup \mathbf{T_3^1}$	$(5,4,2,3,6,0) \sqcup (9,\infty,11)$
	$(4,6,\infty,12,13,1) \sqcup (11,8,7)$
	$(10,\infty,7,6,9,5) \sqcup (16,15,13)$
	$(5,8,10,7,4,11) \sqcup (16,19,17)$
	$(7,0,4,9,15,12) \sqcup (1,11,2)$
$\mathbf{T_6^4} \sqcup \mathbf{T_3^1}$	$(5,4,7,2,1,3) \sqcup (8,11,\infty)$
	$(12,\infty,8,6,4,5) \sqcup (13,10,7)$
	$(10,\infty,2,7,8,5) \sqcup (19,16,14)$
	$(11,10,12,8,5,6) \sqcup (16,13,14)$
	$(7,0,6,4,9,12) \sqcup (1,11,2)$
$\mathbf{T_6^5} \sqcup \mathbf{T_3^1}$	$(1,2,4,5,0,3) \sqcup (8,11,14)$
	$(11,\infty,6,4,8,5) \sqcup (10,13,12)$
	$(6,7,\infty,3,8,5) \sqcup (9,12,15)$
	$(11,10,8,6,12,7) \sqcup (13,16,\infty)$
	$(8,0,4,9,6,7) \sqcup (1,11,2)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_6^6} \sqcup \mathbf{T_3^1}$	$(1,2,0,3,4,5) \sqcup (11,8,\infty)$ $(2,\infty,3,4,5,6) \sqcup (12,13,15)$ $(6,7,8,4,5,\infty) \sqcup (11,12,15)$ $(11,10,8,12,13,7) \sqcup (9,6,4)$ $(4,0,8,5,6,7) \sqcup (1,11,2)$
$\mathbf{T_5^1} \sqcup \mathbf{T_4^1}$	$(5,4,2,\infty,1) \sqcup (11,13,12,15)$ $(0,2,5,\infty,6) \sqcup (8,11,10,12)$ $(5,7,\infty,3,6) \sqcup (16,13,14,15)$ $(\infty,4,7,10,8) \sqcup (17,16,15,13)$ $(4,9,15,8,16) \sqcup (2,11,1,5)$
$\mathbf{T_5^2} \sqcup \mathbf{T_4^1}$	$(\infty,2,4,5,0) \sqcup (11,13,12,15)$ $(6,\infty,5,2,1) \sqcup (8,11,10,12)$ $(6,3,\infty,7,1) \sqcup (16,13,14,15)$ $(10,7,4,\infty,5) \sqcup (17,16,15,13)$ $(16,8,15,9,12) \sqcup (2,11,1,6)$
$\mathbf{T_5^2} \sqcup \mathbf{T_4^2}$	$(\infty,2,4,3,0) \sqcup (11,13,12,15)$ $(6,\infty,5,2,1) \sqcup (10,12,11,15)$ $(6,3,\infty,7,1) \sqcup (12,14,13,15)$ $(\infty,4,7,10,1) \sqcup (17,16,13,15)$ $(16,8,15,9,12) \sqcup (10,1,11,6)$
$\mathbf{T_5^3} \sqcup \mathbf{T_4^1}$	$(0,2,1,3,4) \sqcup (11,8,\infty,6)$ $(2,\infty,3,4,5) \sqcup (9,12,13,15)$ $(4,7,5,6,\infty) \sqcup (11,12,15,14)$ $(0,3,1,5,6) \sqcup (16,13,11,10)$ $(5,1,10,11,6) \sqcup (16,8,15,9)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_5^1} \sqcup \mathbf{T_4^2}$	$(10, 13, \infty, 8, 11) \sqcup (1, 2, 3, 4)$
	$(15, 13, 12, 9, 7) \sqcup (3, \infty, 4, 5)$
	$(11, 12, 15, 14, 13) \sqcup (4, 7, 5, \infty)$
	$(3, 4, 6, 9, \infty) \sqcup (8, 10, 12, 7)$
	$(16, 8, 15, 9, 5) \sqcup (10, 1, 11, 6)$
$\mathbf{T_5^3} \sqcup \mathbf{T_4^2}$	$(0, 2, 3, 4, 5) \sqcup (9, 8, 11, \infty)$
	$(2, \infty, 3, 4, 5) \sqcup (12, 13, 14, 15)$
	$(4, 7, 8, 5, \infty) \sqcup (10, 12, 11, 15)$
	$(0, 3, 1, 4, 6) \sqcup (16, 13, 11, \infty)$
	$(5, 1, 10, 11, 6) \sqcup (16, 8, 14, 15)$
$\mathbf{T_6^1} \sqcup 2\mathbf{T_2^1}$	$(3, 5, 4, 2, \infty, 1) \sqcup (19, 20) \sqcup (12, 15)$
	$(0, 2, 5, \infty, 6, 4) \sqcup (17, 18) \sqcup (8, 11)$
	$(5, 7, \infty, 3, 6, 9) \sqcup (13, 14) \sqcup (0, 1)$
	$(\infty, 4, 7, 10, 8, 6) \sqcup (16, 15) \sqcup (2, 3)$
	$(0, 4, 9, 15, 8, 16) \sqcup (1, 11) \sqcup (3, 12)$
$\mathbf{T_6^2} \sqcup 2\mathbf{T_2^1}$	$(\infty, 2, 4, 5, 8, 0) \sqcup (18, 20) \sqcup (12, 13)$
	$(13, \infty, 5, 2, 3, 8) \sqcup (9, 6) \sqcup (16, 15)$
	$(6, 3, \infty, 7, 5, 4) \sqcup (13, 14) \sqcup (0, 1)$
	$(15, 17, 14, 11, \infty, 19) \sqcup (8, 6) \sqcup (1, 4)$
	$(0, 4, 9, 15, 8, 12) \sqcup (1, 11) \sqcup (5, 14)$
$\mathbf{T_6^5} \sqcup 2\mathbf{T_2^1}$	$(3, 2, 4, 5, 0, 1) \sqcup (18, 15) \sqcup (11, 14)$
	$(5, \infty, 6, 4, 8, 11) \sqcup (10, 13) \sqcup (19, 20)$
	$(8, 7, \infty, 3, 5, 6) \sqcup (16, 19) \sqcup (12, 15)$
	$(7, 10, 8, 6, 11, 12) \sqcup (16, 13) \sqcup (9, \infty)$
	$(6, 0, 8, 4, 5, 7) \sqcup (1, 11) \sqcup (3, 12)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_6^4} \sqcup 2\mathbf{T_2^1}$	$(5,4,7,2,1,3) \sqcup (8,11) \sqcup (18,\infty)$
	$(12,\infty,8,6,4,5) \sqcup (0,3) \sqcup (10,13)$
	$(10,\infty,2,7,8,5) \sqcup (9,6) \sqcup (16,19)$
	$(11,10,12,8,5,6) \sqcup (13,14) \sqcup (0,2)$
	$(7,0,6,4,9,12) \sqcup (1,11) \sqcup (5,14)$
$\mathbf{T_6^3} \sqcup 2\mathbf{T_2^1}$	$(5,4,2,3,6,0) \sqcup (9,12) \sqcup (11,\infty)$
	$(4,6,\infty,12,13,15) \sqcup (0,1) \sqcup (8,11)$
	$(10,\infty,7,6,9,5) \sqcup (13,15) \sqcup (1,2)$
	$(5,8,10,7,4,11) \sqcup (17,19) \sqcup (9,\infty)$
	$(7,0,4,9,15,12) \sqcup (1,11) \sqcup (5,14)$
$\mathbf{T_6^6} \sqcup 2\mathbf{T_2^1}$	$(1,2,0,3,4,5) \sqcup (\infty,15) \sqcup (8,11)$
	$(11,\infty,2,3,5,6) \sqcup (13,15) \sqcup (19,20)$
	$(6,7,8,4,5,\infty) \sqcup (18,19) \sqcup (12,15)$
	$(11,10,8,12,13,7) \sqcup (18,20) \sqcup (9,6)$
	$(11,1,8,9,10,7) \sqcup (0,5) \sqcup (2,6)$
$\mathbf{T_5^1} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(10,13,\infty,8,11) \sqcup (3,2,4) \sqcup (16,15)$
	$(15,13,12,9,7) \sqcup (10,\infty,5) \sqcup (11,14)$
	$(11,12,15,14,13) \sqcup (4,\infty,7) \sqcup (0,3)$
	$(3,4,6,9,\infty) \sqcup (8,10,12) \sqcup (5,7)$
	$(0,9,1,8,2) \sqcup (5,10,6) \sqcup (3,13)$
$\mathbf{T_5^2} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(8,\infty,13,10,9) \sqcup (3,2,4) \sqcup (14,15)$
	$(7,9,12,13,8) \sqcup (10,\infty,5) \sqcup (11,14)$
	$(11,12,15,18,14) \sqcup (4,\infty,7) \sqcup (0,3)$
	$(9,6,4,3,8) \sqcup (19,17,15) \sqcup (13,14)$
	$(1,8,0,9,2) \sqcup (5,10,6) \sqcup (3,13)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_5^3} \sqcup \mathbf{T_3^1} \sqcup \mathbf{T_2^1}$	$(2, \infty, 3, 4, 5) \sqcup (12, 13, 15) \sqcup (16, 19)$ $(0, 2, 1, 3, 4) \sqcup (8, \infty, 6) \sqcup (18, 15)$ $(4, 7, 5, 6, \infty) \sqcup (11, 12, 15) \sqcup (0, 1)$ $(8, 10, 12, 13, 7) \sqcup (9, 6, 4) \sqcup (17, 18)$ $(9, 0, 8, 6, 7) \sqcup (11, 1, 5) \sqcup (10, 15)$
$2\mathbf{T_4^1} \sqcup \mathbf{T_2^1}$	$(1, \infty, 16, 18) \sqcup (11, 13, 12, 15) \sqcup (4, 5)$ $(2, 5, \infty, 6) \sqcup (8, 11, 10, 12) \sqcup (9, 7)$ $(0, \infty, 3, 6) \sqcup (16, 13, 14, 15) \sqcup (5, 7)$ $(10, 7, 4, \infty) \sqcup (17, 16, 15, 13) \sqcup (1, 3)$ $(9, 15, 8, 16) \sqcup (2, 11, 1, 5) \sqcup (12, 7)$
$\mathbf{T_4^1} \sqcup \mathbf{T_4^2} \sqcup \mathbf{T_2^1}$	$(11, 9, \infty, 1) \sqcup (10, 12, 13, 15) \sqcup (4, 5)$ $(2, 5, \infty, 6) \sqcup (8, 11, 10, 13) \sqcup (9, 7)$ $(0, \infty, 17, 20) \sqcup (12, 14, 13, 15) \sqcup (8, 6)$ $(10, 7, 4, \infty) \sqcup (17, 16, 13, 15) \sqcup (1, 3)$ $(2, 12, 6, 15) \sqcup (8, 0, 5, 7) \sqcup (9, 13)$
$2\mathbf{T_4^2} \sqcup \mathbf{T_2^1}$	$(18, 16, 19, \infty) \sqcup (10, 12, 13, 15) \sqcup (3, 6)$ $(1, \infty, 12, 6) \sqcup (8, 11, 10, 13) \sqcup (4, 5)$ $(0, \infty, 3, 4) \sqcup (12, 14, 13, 15) \sqcup (8, 6)$ $(9, 7, 10, 4) \sqcup (17, 16, 13, 15) \sqcup (1, 3)$ $(9, 0, 8, 7) \sqcup (11, 1, 5, 6) \sqcup (10, 4)$
$\mathbf{T_4^1} \sqcup 2\mathbf{T_3^1}$	$(11, 13, 12, 15) \sqcup (9, \infty, 1) \sqcup (2, 4, 5)$ $(8, 11, 10, 12) \sqcup (19, \infty, 6) \sqcup (0, 2, 5)$ $(0, \infty, 3, 6) \sqcup (16, 13, 14) \sqcup (8, 7, 5)$ $(17, 16, 15, 13) \sqcup (\infty, 4, 7) \sqcup (0, 3, 1)$ $(9, 15, 8, 16) \sqcup (11, 1, 5) \sqcup (3, 12, 7)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_4^2} \sqcup 2\mathbf{T_3^1}$	$(18,16,19,\infty) \sqcup (13,12,15) \sqcup (5,3,6)$ $(1,\infty,12,6) \sqcup (8,11,13) \sqcup (3,4,5)$ $(0,\infty,3,4) \sqcup (12,14,13) \sqcup (6,8,7)$ $(9,7,10,4) \sqcup (17,16,13) \sqcup (2,1,3)$ $(9,0,8,7) \sqcup (5,1,6) \sqcup (10,4,14)$
$\mathbf{T_4^1} \sqcup \mathbf{T_3^1} \sqcup 2\mathbf{T_2^1}$	$(11,13,12,15) \sqcup (9,\infty,1) \sqcup (4,5) \sqcup (16,18)$ $(8,11,10,12) \sqcup (19,\infty,6) \sqcup (2,5) \sqcup (16,14)$ $(8,10,7,4) \sqcup (0,\infty,11) \sqcup (16,17) \sqcup (9,6)$ $(5,7,8,6) \sqcup (20,17,\infty) \sqcup (13,14) \sqcup (1,2)$ $(3,10,5,11) \sqcup (0,9,1) \sqcup (2,12) \sqcup (17,13)$
$\mathbf{T_4^2} \sqcup \mathbf{T_3^1} \sqcup 2\mathbf{T_2^1}$	$(18,16,19,\infty) \sqcup (13,12,15) \sqcup (3,5) \sqcup (17,20)$ $(1,\infty,12,6) \sqcup (8,11,13) \sqcup (4,5) \sqcup (17,18)$ $(3,\infty,4,7) \sqcup (12,14,13) \sqcup (8,6) \sqcup (1,2)$ $(9,7,10,4) \sqcup (17,16,13) \sqcup (1,3) \sqcup (14,15)$ $(9,0,8,7) \sqcup (11,1,6) \sqcup (18,12) \sqcup (10,14)$
$\mathbf{T_5^1} \sqcup 3\mathbf{T_2^1}$	$(4,1,\infty,13,10) \sqcup (2,3) \sqcup (16,15) \sqcup (9,11)$ $(5,\infty,10,11,13) \sqcup (4,7) \sqcup (0,2) \sqcup (9,12)$ $(7,\infty,4,5,8) \sqcup (17,19) \sqcup (0,3) \sqcup (12,14)$ $(7,8,6,9,\infty) \sqcup (13,14) \sqcup (1,3) \sqcup (19,20)$ $(1,11,2,10,3) \sqcup (0,6) \sqcup (9,4) \sqcup (8,12)$
$\mathbf{T_5^2} \sqcup 3\mathbf{T_2^1}$	$(1,\infty,13,10,7) \sqcup (2,3) \sqcup (16,15) \sqcup (9,11)$ $(5,\infty,10,11,16) \sqcup (4,7) \sqcup (0,2) \sqcup (9,12)$ $(6,4,5,8,\infty) \sqcup (17,19) \sqcup (0,3) \sqcup (12,14)$ $(7,8,6,9,11) \sqcup (13,14) \sqcup (1,3) \sqcup (19,20)$ $(3,10,2,11,5) \sqcup (0,6) \sqcup (4,8) \sqcup (17,7)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_5^3} \sqcup 3\mathbf{T_2^1}$	$(1, \infty, 13, 5, 7) \sqcup (2, 3) \sqcup (16, 15) \sqcup (9, 11)$ $(0, 3, 1, 4, \infty) \sqcup (2, 5) \sqcup (9, 7) \sqcup (10, 13)$ $(12, 11, 13, 14, \infty) \sqcup (17, 19) \sqcup (5, 7) \sqcup (9, 6)$ $(5, 8, 11, 6, 7) \sqcup (13, 14) \sqcup (2, \infty) \sqcup (19, 20)$ $(6, 0, 8, 9, 7) \sqcup (1, 11) \sqcup (10, 5) \sqcup (16, 12)$
$\mathbf{T_4^1} \sqcup 2\mathbf{T_3^1}$	$(11, 13, 12, 15) \sqcup (9, \infty, 1) \sqcup (2, 4, 5)$ $(8, 11, 10, 12) \sqcup (19, \infty, 6) \sqcup (0, 2, 5)$ $(0, \infty, 3, 6) \sqcup (16, 13, 14) \sqcup (8, 7, 5)$ $(17, 16, 15, 13) \sqcup (\infty, 4, 7) \sqcup (0, 3, 1)$ $(9, 15, 8, 16) \sqcup (11, 1, 5) \sqcup (3, 12, 7)$
$\mathbf{T_4^1} \sqcup 4\mathbf{T_2^1}$	$(9, \infty, 8, 6) \sqcup (12, 15) \sqcup (16, 17) \sqcup (1, 2) \sqcup (19, 20)$ $(5, \infty, 13, 14) \sqcup (9, 6) \sqcup (0, 2) \sqcup (1, 4) \sqcup (17, 19)$ $(0, \infty, 4, 3) \sqcup (10, 7) \sqcup (16, 18) \sqcup (2, 5) \sqcup (11, 14)$ $(18, 20, 17, \infty) \sqcup (4, 5) \sqcup (12, 14) \sqcup (8, 10) \sqcup (0, 1)$ $(0, 9, 1, 11) \sqcup (10, 3) \sqcup (12, 6) \sqcup (19, 14) \sqcup (17, 13)$
$\mathbf{T_4^2} \sqcup 4\mathbf{T_2^1}$	$(8, \infty, 9, 5) \sqcup (12, 15) \sqcup (16, 17) \sqcup (1, 2) \sqcup (3, 4)$ $(15, 13, 14, \infty) \sqcup (9, 6) \sqcup (0, 2) \sqcup (1, 4) \sqcup (17, 19)$ $(0, \infty, 3, 4) \sqcup (10, 7) \sqcup (16, 18) \sqcup (2, 5) \sqcup (11, 14)$ $(17, 20, 18, 19) \sqcup (4, 5) \sqcup (12, 14) \sqcup (8, 10) \sqcup (0, 1)$ $(9, 0, 8, 7) \sqcup (1, 11) \sqcup (12, 6) \sqcup (10, 5) \sqcup (16, 20)$
$2\mathbf{T_3^1} \sqcup 3\mathbf{T_2^1}$	$(8, \infty, 9) \sqcup (13, 12, 15) \sqcup (4, 5) \sqcup (16, 18) \sqcup (1, 2)$ $(19, \infty, 6) \sqcup (11, 10, 12) \sqcup (2, 5) \sqcup (18, 20) \sqcup (1, 4)$ $(11, \infty, 14) \sqcup (10, 7, 4) \sqcup (16, 17) \sqcup (0, 2) \sqcup (1, 3)$ $(20, 17, \infty) \sqcup (14, 13, 15) \sqcup (5, 7) \sqcup (9, 6) \sqcup (0, 1)$ $(0, 9, 4) \sqcup (2, 10, 3) \sqcup (12, 6) \sqcup (17, 7) \sqcup (1, 5)$

Table A.2: 1-rotational (1-2-3)-labelings.

Forest	Labeling
$\mathbf{T_3^1} \sqcup 5\mathbf{T_2^1}$	$(8,\infty,9) \sqcup (12,15) \sqcup (4,5) \sqcup (16,18) \sqcup (1,2) \sqcup (19,20)$ $(5,\infty,13) \sqcup (9,6) \sqcup (0,2) \sqcup (18,20) \sqcup (1,4) \sqcup (17,19)$ $(11,\infty,14) \sqcup (4,7) \sqcup (16,17) \sqcup (2,5) \sqcup (8,10) \sqcup (0,3)$ $(20,17,\infty) \sqcup (13,14) \sqcup (5,7) \sqcup (10,11) \sqcup (0,1) \sqcup (8,6)$ $(0,9,4) \sqcup (2,10,3) \sqcup (12,6) \sqcup (17,7) \sqcup (1,5)$

A.3 $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$-decompositions of K_{21} and K_{22}

Table A.3: A $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$-decomposition of K_{21}

No.	Block	No.	Block
1	$(15,14,16,17,18,19,20) \sqcup (0,2)$	2	$(13,15,16,17,18,19,20) \sqcup (0,6)$
3	$(8,16,12,17,18,19,20) \sqcup (9,3)$	4	$(8,17,9,11,18,19,20) \sqcup (16,0)$
5	$(8,18,9,11,13,19,20) \sqcup (0,1)$	6	$(8,19,10,11,12,13,20) \sqcup (0,15)$
7	$(8,1,9,10,11,12,13) \sqcup (18,7)$	8	$(1,2,9,10,11,12,13) \sqcup (14,7)$
9	$(0,3,2,6,11,12,13) \sqcup (8,7)$	10	$(0,4,2,3,11,12,13) \sqcup (8,9)$
11	$(0,5,2,3,4,12,13) \sqcup (9,10)$	12	$(1,6,2,4,5,12,13) \sqcup (15,7)$
13	$(1,7,2,3,4,5,6) \sqcup (0,14)$	14	$(3,8,4,5,6,14,20) \sqcup (12,15)$
15	$(4,9,5,6,14,15,20) \sqcup (16,7)$	16	$(15,10,4,5,6,16,20) \sqcup (0,18)$
17	$(15,11,0,5,6,16,20) \sqcup (17,1)$	18	$(14,12,0,11,17,18,20) \sqcup (8,2)$
19	$(16,13,0,11,12,17,20) \sqcup (1,19)$	20	$(1,14,2,3,4,5,6) \sqcup (20,7)$
21	$(1,15,2,3,4,5,6) \sqcup (19,7)$	22	$(1,16,2,3,4,5,6) \sqcup (17,7)$
23	$(0,17,2,3,4,5,6) \sqcup (11,14)$	24	$(1,18,2,3,4,5,6) \sqcup (10,14)$
25	$(0,19,2,3,4,5,6) \sqcup (13,14)$	26	$(0,20,2,3,4,5,6) \sqcup (10,11)$
27	$(9,7,0,10,11,12,13) \sqcup (1,3)$	28	$(10,8,0,11,12,13,15) \sqcup (1,4)$
29	$(11,9,0,12,13,16,19) \sqcup (1,5)$	30	$(12,10,0,3,13,17,18) \sqcup (1,20)$

Table A.4: A $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^1}$-decomposition of K_{22}

No.	Block	No.	Block
1	$(15,14,16,17,18,19,20) \sqcup (0,2)$	2	$(13,15,16,17,18,19,20) \sqcup (0,6)$
3	$(8,16,12,17,18,19,20) \sqcup (9,3)$	4	$(8,17,9,11,18,19,20) \sqcup (16,0)$
5	$(8,18,9,11,13,19,20) \sqcup (0,1)$	6	$(8,19,10,11,12,13,20) \sqcup (0,15)$
7	$(8,1,9,10,11,12,13) \sqcup (6,\infty)$	8	$(1,2,9,10,11,12,13) \sqcup (14,7)$
9	$(0,3,2,6,11,12,13) \sqcup (8,7)$	10	$(0,4,2,3,11,12,13) \sqcup (8,9)$
11	$(0,5,2,3,4,12,13) \sqcup (9,10)$	12	$(1,6,2,4,5,12,13) \sqcup (15,7)$
13	$(1,7,2,3,4,5,6) \sqcup (13,\infty)$	14	$(3,8,4,5,6,14,20) \sqcup (12,15)$

Table A.4: A $\mathbf{T_7^{11}} \sqcup \mathbf{T_2^{1}}$-decomposition of K_{22}

No.	Block	No.	Block
15	$(4, 9, 5, 6, 14, 15, 20) \sqcup (16, 7)$	16	$(15, 10, 4, 5, 6, 16, 20) \sqcup (0, 18)$
17	$(15, 11, 0, 5, 6, 16, 20) \sqcup (17, 1)$	18	$(14, 12, 0, 11, 17, 18, 20) \sqcup (8, 2)$
19	$(16, 13, 0, 11, 12, 17, 20) \sqcup (1, 19)$	20	$(1, 14, 2, 3, 4, 5, 6) \sqcup (20, 7)$
21	$(1, 15, 2, 3, 4, 5, 6) \sqcup (19, 7)$	22	$(1, 16, 2, 3, 4, 5, 6) \sqcup (17, 7)$
23	$(0, 17, 2, 3, 4, 5, 6) \sqcup (11, 14)$	24	$(1, 18, 2, 3, 4, 5, 6) \sqcup (10, 14)$
25	$(0, 19, 2, 3, 4, 5, 6) \sqcup (13, 14)$	26	$(0, 20, 2, 3, 4, 5, 6) \sqcup (10, 11)$
27	$(9, 7, 0, 10, 11, 12, 13) \sqcup (20, \infty)$	28	$(10, 8, 0, 11, 12, 13, 15) \sqcup (1, 4)$
29	$(11, 9, 0, 12, 13, 16, 19) \sqcup (1, 5)$	30	$(12, 10, 0, 3, 13, 17, 18) \sqcup (1, 20)$
31	$(0, \infty, 1, 2, 3, 4, 5) \sqcup (18, 7)$	32	$(14, \infty, 15, 16, 17, 18, 19) \sqcup (1, 3)$
33	$(7, \infty, 8, 9, 10, 11, 12) \sqcup (0, 14)$		