2006

This is to certify that the

dissertation entitled

ON GRAPHS WITH LARGE NUMBERS OF SPANNING TREES

presented by

Andrew Chen

has been accepted towards fulfillment
of the requirements for

Ph.D degree in Computer Science

Major professor

Date July 11, 2005

PLACE IN RETURN BOX to remove this checkout from your record.
TO AVOID FINES return on or before date due.
MAY BE RECALLED with earlier due date if requested.

DATE DUE	DATE DUE	DATE DUE

On Graphs with Large Numbers of Spanning Trees

By

Andrew Chen

A Dissertation

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

Department of Computer Science

2005

Abstract

On Graphs with Large Numbers of Spanning Trees

By

Andrew Chen

Communication network topology designs modeled by graphs with a large number of spanning trees tend towards higher reliability. Given the number of vertices n and the number of edges m, we have studied graphs with a large number of spanning trees and have characterized (n, m) graphs that have maximum number of spanning trees. We studied $(n, n+4)$ graphs, a formerly open case of this problem, as well as other aspects of this problem.

Let $t(G)$ denote the number of labeled spanning trees of a connected graph G. Given G, it is known how to compute $t(G)$. However, little is known about the extremal version of the problem, that is, given the number of vertices n and the number of edges m, find a connected (n, m) graph G such that $t(G) \geq t(H)$, where H is any other (n, m) connected graph. Such a graph G is called a t-optimal graph. We present a summary of the known results on t-optimal graphs and we present some new results for the case of $(n, n+4)$ t-optimal graphs and begin to develop techniques for exploring (n, m) graphs in general.

Let $t(n, m)$ be the number of spanning trees that a t-optimal (n, m) graph has. We present (obtained through using a software called *nauty*) the values of $t(n, m)$ for $n \leq 12$. We also present histograms of $t(G)$ for $G \in (n, n+4)$ for $n < 14$. Additionally we present information about the few non-isomorphic t-optimal graphs that we have

encountered.

The Feussman formula provides a well-known recursive relationship for the number of spanning trees of a graph G in terms of the deletion and contraction of an edge e, namely, $t(G) = t(G - e) + t(G \cdot e)$. A common technique for finding the number of spanning trees of a graph is finding the minor of a matrix form of a graph. However, the impact of relatively simple changes to a graph (such as edge contraction or edge deletion) is not readily visible when the technique of finding the minor of a matrix form of a graph is used.

We show a technique for computing the number of spanning trees of a graph which clearly shows the impact of graph operations such as edge deletion, subdivision, and contraction on the number of spanning trees of a graph. Additionally we extend that result to include edge addition. Our technique makes use of reductions of the graph by repeatedly contracting cycles, and is demonstrated to be advantageous to the current techniques for the computation of $t(G)$ when the modeling of such graph operations is desired.

ACKNOWLEDGMENTS

"If I have seen further it is by standing on ye shoulders of Giants" - Isaac Newton

First and foremost, I could not have made the progress I did on this problem if it had not been for those that have worked on it and in this field before. They are too numerous to mention, but a small sampling of them can be found in the bibliography.

Gratitude is due to Professor McKay for making the *nauty* software [1] available. For every other piece of software, there may have been a (perhaps more time consuming) alternative that could have been used, but *nauty* exists in a class unto itself.

Without the suggestion from my advisor, Dr. Esfahanian, that I consider this problem, I might very well have bitten off far more than I could chew by pursuing another problem, so great thanks is due to him, not only for suggesting the problem, but for great understanding, patience, and wise counsel with regard to both this and other things.

Friends and acquaintances far too numerous to mention have been very encouraging in both reminding me to work on this and in reminding me to take a break from it. For those who gave me appropriate amounts of encouragement of each, I thank them greatly. Connie deserves special mention, as do the residents of Owen Graduate Center, the Society of Success and Leadership's 2004-2005 Success Networking Team, and the campus ministry of The People's Church. All of these groups, the people in them, and others helped me to cultivate the appropriate mindset to carry me through

this task and retain my sanity.

The task of listing everyone that helped is an arduous one - I am tempted to dump my entire address book here. That might be excessive.

I wouldn't feel right if I didn't mention my parents, whose concern for my progress was genuinely felt, and that higher power which no words can completely describe.

Table of Contents

List of Figures

LIST OF TABLES

Chapter 1

Introduction

Communication network topology designs modeled by graphs with a large number of spanning trees tend towards higher reliability [2]. Graphs with the most trees give the best topology when the edges are sufficiently unreliable [3]. This makes intuitive sense due to the fact that the number of spanning trees is the number of ways we can have a maximal number of link failures in the network while preserving connectivity.

A network can be modeled by a graph $G = (V, E)$, where V is the vertex set and E is the edge set[1]. A tree is a connected graph that would become disconnected if any edge were removed. A spanning tree of a graph is a subgraph of G which is a tree and includes all vertices of G.

We are addressing the problem of finding graphs with maximum number of spanning trees. Given a graph, we know how to find the number of spanning trees of that graph. Given a number of vertices and a number of edges, what graph has those numbers and has the maximum number of spanning trees? This question is the problem we are addressing.

[1]An edge is often thought of as "connecting" two vertices. We are describing undirected edges and we are allowing multiple edges between the same two vertices - also known as parallel edges. We are also allowing multiple edges between the same vertex - also known as loops. If a graph has no parallel edges or loops the graph is known as a simple graph. Technically speaking, E, the set of edges, is a multi-set of unordered subsets of size 2 of V.

Sometimes graphs that are not simple are called multigraphs. In our work we are looking for t-optimal simple graphs, with the realization that certain graph operations may result in multi-graphs.

For graph $G = (V, E)$, G is called dense if $m = O(n^2)$, and sparse if $m = O(n)$, where $m = |E|$ and $n = |V|$. Much work has been done ([4], [5], [6], [7], and [8]) regarding dense cases, but little work has been done regarding most sparse cases. We have focused on the sparsest case which had not been done prior to our work, and we continue to investigate alternative aspects of this problem.

1.1 Problem Statement

Let $t(G)$ be the number of possible labeled spanning trees in a connected graph G. A graph with n vertices and m edges is called an (n, m) graph. An (n, m) graph G is said to be t-optimal if and only if $t(G) \geq t(H)$ where H is any other (n, m) graph. We study the problem of finding t-optimal (n, m) graphs, The case of $m \leq n + 3$ has been previously studied; we studied $(n, n + 4)$ graphs, a formerly open case of this problem, as well as other aspects of this problem.

As part of our studying of this problem, we provide a summary of known results, including many known conjectures. This problem is one for which there are a variety of long-standing conjectures, consideration of which are useful for illuminating the search for solutions.

We sought to determine if, for every (n, m), there is one and only one non-isomorphic t-optimal graph. We also sought to determine the impact of simple operations such as edge deletion, subdivision, contraction, and addition upon the number of spanning trees and the possible preservation of the property of t-optimality. These questions and many more we seek to answer herein.

1.2 Definitions And Notations

We will use $S(n, m)$ to denote the the set of all t-optimal (n, m) graphs. The number of spanning trees of graphs in $S(n, m)$ will be denoted by $t(n, m)$.

To *contract* an edge is to replace an edge and its end vertices with a single new vertex, and any other edges that were incident to either of those two vertices will be changed to instead be incident with the new vertex. The graph obtained by contracting the edge e in G is denoted by $G \cdot e$.

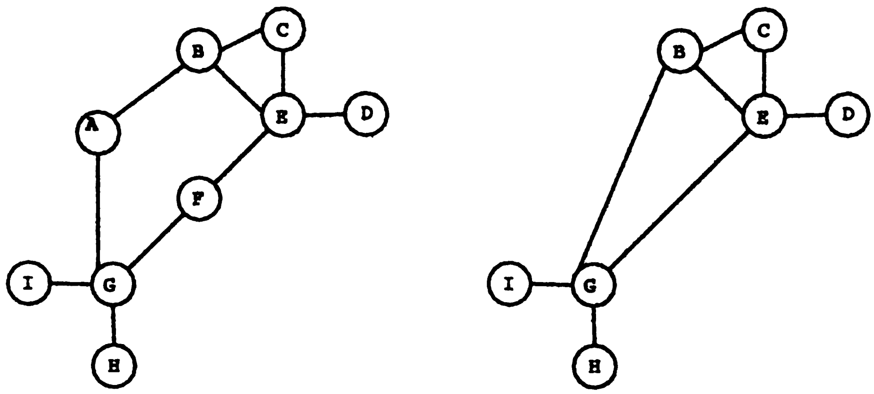
To *subdivide* an edge uv in a graph G is to create a new graph G' by replacing the edge uv with the path $u - w - v$, where w is a new vertex. Contraction may be thought of as the inverse of subdivision.

A *loop* is an edge with identical end vertices. Potentially after a contraction of an edge, a loop may be produced.

Edges that share the same end vertices are called *parallel* edges. Potentially after a contraction, parallel edges may be produced.

To contract a cycle is to contract all the edges in the cycle. We denote the contraction of cycle c in graph G as $G \cdot c$. For our purposes, loops count as cycles, as does a single pair of parallel edges. A formal definition of how we use the term cycle is as follows: A cycle is a sequence of edges and their incident vertices $\{e_0, v_0, e_1, v_1, \ldots, e_h, v_h\}$ such that e_i is incident to v_{i-1} and vi, e_0 is incident to v_0 and v_h, and no vertex occurs in the sequence more than once. Thus, a loop is a cycle because it is a single edge and a single vertex, whereas a single pair of parallel edges is a cycle because it is two edges and two vertices that can be arranged in such a sequence. Figure 4.3(b) is an example of Figure 4.3(a) with the cycle ABE contracted.

A *chain* in a graph G is a path in G in which every internal vertex in the path has degree exactly 2 and the two end vertices of the path have degree strictly greater than two ([9]). An example of a graph with some chains can be found in Figure 1.1(a). One chain in that graph is $g - a - b$; another is $g - f - e$. An example of a path in that graph that is not a chain is $a - b - c - e$ because b is not a vertex of degree 2 and a is not a vertex of degree greater than 2. Another example of a path in that graph that is not a chain is $e - d$ because d is not a vertex of degree greater than 2. An



(a) A graph

(b) An example of a graph that is **not** the distillation of Figure 1.1(a)

Figure 1.1: Examples of graphs that are not distillations

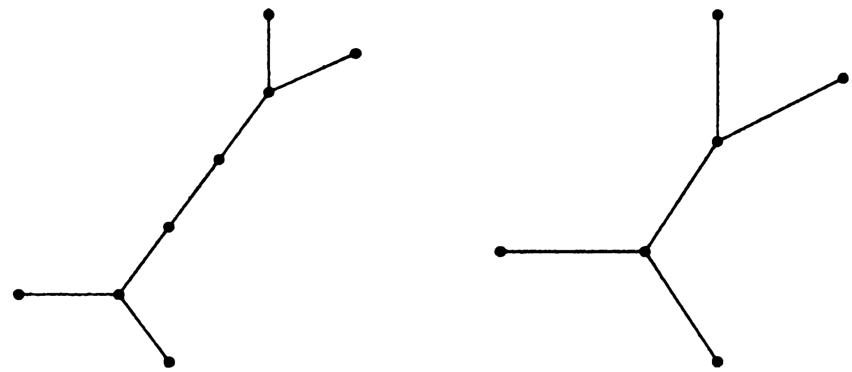
edge whose end vertices are of degree three or greater is a *trivial* chain, an example of which is $b - e$. If two edge-disjoint chains share the same end vertices then they are *parallel* chains. The length of a chain is the number of edges in it.

For a graph G, the *distillation* $D(G)$ of G is the result of repeatedly contracting any edge incident to a vertex of degree 2 until there are no vertices of degree two ([9]). An example of the distillation of the graph in Figure 1.2(a) can be found in Figure 1.2(b).

The distillation of a graph can be seen as replacing every chain with a single edge. Figure 1.1(b) is not a distillation of Figure 1.1(a) because vertex c has degree 2. The distillation of Figure 1.1(a) would have two parallel trivial chains between b and e and no other chains between b and e.

If we take the distillation of a graph, we can subdivide some number of edges some number of times to recreate the original graph. When we do this, we are merely changing the lengths of the chains.

The cyclespace $C(G)$ of a graph G is the set of all cycles that are subgraphs of G. If B is a basis for the cyclespace of G then B is a least sized set of cycles such that



(a) An example of a graph with a non-trivial chain

(b) An example of a graph with only trivial chains, and the distillation of Figure 1.2(a)

Figure 1.2: Examples of trivial and non-trivial chains

every cycle in $C(G)$ can be generated by the symmetric difference of some subset of B. The dimension of a cyclespace is the size of B. The dimension of the cyclespace of a graph G is sometimes called the cycle rank of G. The cycle rank [10] of an (n, m) graph G is $m - n + \omega$ where ω is the number of components in G. Thus the cycle rank of a connected (n, m) graph G is $m - n + 1$ as there is only one component.

A common technique for finding $t(G)$ is finding the minor of a matrix form of G, known as the *Kirchoff matrix*. (The minor of a matrix is the determinant of the matrix after the removal of a row and a column.) In other literature, the *Kirchoff matrix* is sometimes referred to as the *Laplacian* or *admittance matrix*, as found in [8]. An example of a Kirchoff matrix can be found in Figure 4.7.

In [9], several useful theorems are provided, including a reference to the Feussner formula [11] which is

$$t(G) = t(G - e) + t(G \cdot e)$$

where e is any edge in G which is not a loop. This formula makes intuitive sense: it is the sum of those spanning trees that do not include e with those that do include e. If e is a loop then $t(G) = t(G - e) = t(G \cdot e)$ because e can not be in a spanning tree if it is a loop.

A multipartite graph has its vertex set partitioned into p subsets called partitions. There are no edges joining vertices within a partition. A complete multipartite graph is one in which there is always an edge between any two vertices in different partitions.

A complete r-regular multipartite graph can be completely described by two numbers, namely k and p, where k is the number of vertices in a partition, and p is the number of partitions. There is an edge between any two vertices not in the same partition, and there are no edges between vertices in the same partition. We have $n = kp$, $r = k(p-1)$, and

$$m = n(r/2) = k^2\binom{p}{2},$$

where r is the degree of a vertex. If we consider $m - n$, we get

$$m - n = (n/2)(r-2).$$

If r is 2, we have the case of the single cycle ($m = n$). If r is $n-1$ then

$$m - n = \binom{n}{2} - n$$

and we have the complete graph. So, different values of r cover a variety of ranges of possible values of m and n, but not all. In fact, the values of m and n which are not covered are those values for which no complete regular multipartite graph exists. Figure A.1 is an example of a complete regular graph. Figure A.2 and Figure A.3 are examples of complete regular multipartite graphs.

The degree of a vertex is the number of incident edges. Amongst all the vertices in a graph, the degree of the vertex of greatest degree is referred to as $\Delta(G)$, and the degree of the vertex of least degree is referred to as $\delta(G)$.

An almost regular graph is a graph for which the degrees of no two vertices differ by more than one, alternately phrased as $\Delta(G) - \delta(g) \leq 1$. Thus, a complete multipartite almost regular graph is a graph with p partitions, no edges within a partition, edges between any two vertices in different partitions, and vertex degrees differing by at most one. Note that this is equivalent to being a complete multipartite graph in which partition sizes differ by at most one. An example of a complete multipartite almost regular graph can be found in Figure A.4.

1.3 Executive Summary of Contributions

In our study of this problem we have accomplished many things, including: summarizing the known results on t-optimal graphs, finding some new results for the case of $(n, n+4)$ t-optimal graphs, and beginning to develop techniques for exploring (n, m) graphs in general.

In our investigation of the problem we obtained, with the help of a software called *nauty*, the values of $t(n, m)$ for $n \leq 12$ and histograms of $t(G)$, $G \in (n, n+4)$ for $n < 14$. We have found a few non-isomorphic t-optimal graphs.

For every distillation there is an expression for the number of spanning trees in terms of the chain lengths, this we have proven. We describe how to find these expressions and the advantages of working with them as opposed to the matrix technique for finding the number of spanning trees.

It is known that for each k, such that $k \leq 3$ and $k \leq \frac{3n}{2}$, all graphs in $S(n, n+k)$ have the same distillation ([12],[13], [9] and [14]). This yields a method of finding $(n, n+k)$ t-optimal graphs when $k \leq 3$, by taking a "seed" graph (dependent on k) and subdividing its edges in a specific order. We have formulated the problem of finding t-optimal $(n, n+k)$ graphs as a polynomial integer optimization problem.

Using the method of subdividing edges of a "seed" and the polynomial integer optimization formulation, we have found a "seed" graph that gives an infinite number

of members of $S(n, n+4)$. It has been conjectured [15] that all graphs in $S(n, n+k)$ have the same distillation for $k \leq \frac{n}{2}$. Within that range we also present a heuristic for generating graphs that tend to have large number of spanning trees. As a consequence of some of our work, we also prove that t-optimal graphs do not have distillations in which there is an edge whose removal results in a graph that is not biconnected.

The impact of relatively simple changes to a graph (such as edge contraction or edge deletion) is not readily visible when the technique of finding the minor of a matrix form of a graph is used. The expression for the number of spanning trees, in terms of the chain lengths, provides us with a technique for computing the number of spanning trees of a graph which shows clearly the impact of graph operations such as edge deletion, subdivision, and contraction on the number of spanning trees of a graph. We then extended that result to include edge addition. We make use of reductions of the graph by repeatedly contracting cycles in this technique.

1.4 Overview

This chapter (Chapter 1) serves as an introduction to the problem (Section 1.1), provides some definitions and notations that are used throughout the remainder of this work (Section 1.2), provides an executive summary of the contributions found in this work (Section 1.3), and provides an overview of the other chapters (Section 1.4). The next chapter (Chapter 2) provides an overview of the previous work that has been done by others on this problem. Following that we have Chapter 3 which describes some of the preliminary results that had been obtained as of the time we formally proposed pursuing this avenue of research. The newest results can be found in Chapter 4. Some avenues to continue pursuit of this problem are mentioned in Chapter 5.

Chapter 2

Previous Work

In this section we outline some of the previous work that others have done on this problem. This problem is one for which there are a variety of long-standing conjectures, consideration of which are useful for illuminating the search for solutions. Section 2.1 provides some known results including many of the longstanding conjectures pertaining to this problem. Section 2.2 focuses on results pertaining to dense graphs and Section 2.3 focuses on results pertaining to sparse graphs. A simple overview of some of ranges for which this problem has been solved can be found in Section 2.4.

2.1 General Theorems

We have an upper bound for $t(n, m)$ in terms of the number of vertices (Cayley's Theorem [16] for complete graphs) which is n^{n-2}. Since a spanning tree consists of $n - 1$ edges, $\binom{m}{n-1}$ is an upper bound as well. If $m = n - 1$ we can have at most one spanning tree, and hence that number is the maximum number of spanning trees. If $m = n$ then considering the $\binom{m}{n-1}$ upper bound, we see that we have n as an upper bound for $t(n, n)$. A simple cycle achieves this upper bound because the removal of any edge creates a spanning tree and thus $n = t(n, n)$. When n is fixed,

increasing m can only increase the number of spanning trees. Thus we have bounds $n \leq t(n,m) \leq n^{n-2}$ if $m \geq n$.

The Feussner formula was used substantially in the contributions of Boesch et. al. in "On the existence of uniformly optimally reliable networks" [9]; those contributions are fundamental to the approach we take. They prove that for any t-optimal (n,m) graph G, if $m \geq n \geq 3$ then G is biconnected [9] where by biconnected we mean that between any two vertices there are at least two vertex-disjoint paths. In that same work Boesch et. al. also prove that if an (n,m) graph G is t-optimal and $m \geq n+2 \geq 6$ and G has two parallel chains then both chains are trivial [9].

The *Kirchoff matrix* technique has been applied most successfully to specific classes of graphs with known structure. The square of a graph G is defined as follows: H is the square of a graph G if $V(H) = V(G)$ and $E(H) = \{e = uv | d(u,v) \leq 2\}$ where $d(u,v)$ is the length of a shortest path between u and v in G. In [17] the technique of the *Kirchoff matrix* is applied to the square of a cycle to determine that the number of spanning trees in the square of a cycle is $n{F_n}^2$ (where F_n is the nth Fibonacci number). Presumably this technique could be applied to other graphs that have the same kind of symmetry as the square of a cycle does, for example as in [18]. Hitherto, the sort of symmetry for which this technique is easily applicable has been elusive.

We can get more precise bounds by concentrating on the dense or the sparse cases. Gilbert and Myrvold in [15] have a recent summary of research done on both the sparse and dense cases, and they have Conjecture 1, Conjecture 2, and Conjecture 3 as follows.

Conjecture 1 *t-optimal graphs are almost regular*

Conjecture 2 *t-optimal graphs do not have multiple edges as long as*

$$m \leq \binom{n}{2}$$

Conjecture 3 *t-optimal graphs with maximum degree three and the same number of vertices of degree three have the same distillation*

Some statistics work done in the field of design of experiments by Cheng in [19] was re-interpreted for graph theory in [20]. The relevant result is all complete r-regular multi-partite graphs are t-optimal.

Petingi et. al. [21] prove that Conjecture 1 is asymptotically true. Petingi et. al. [8] also prove that complete multipartite almost regular graphs are t-optimal.

2.2 The Case of Dense Graphs

If m is $\binom{n}{2}$ then $t(n, m) = n^{n-2}$. If m is in the range

$$\binom{n}{2} - \frac{n}{2} \leq m \leq \binom{n}{2}$$

then the graph G with maximum number of spanning trees has, as a complement, a set of independent edges ([4] and also in [5]).

Petingi et. al. have done a number of other dense cases ([6] and [7]) extending the cases all the way through $\binom{n}{2} - n$. Gilbert and Myrvold in [15] extend the range of dense graphs covered, subject to Conjecture 1.

For values of n above a certain constant, Petingi et. al. [8] handle the range

$$\binom{n}{2} - \frac{3n}{2} \leq m \leq \binom{n}{2} - n$$

So in summary, the range

$$\binom{n}{2} - \frac{3n}{2} \leq m \leq \binom{n}{2}$$

has been covered, largely by Petingi et. al., and for all but a finite number of values of n, for simple graphs.

2.3 The Case of Sparse Graphs

The sparse cases have crept along. The $m = n - 1$ and $m = n$ cases have been described in Section 2.1. If $m = n + 1$ the graph with the maximum number of spanning trees is a connected graph with two vertices of degree three, and all other vertices of degree two (looks like a θ) as described in [12]. The $m = n + 2$ case was handled in [13] and the solution is when the graph is a particular subdivision of K_4. In [9] the $m = n+3$ case is stated as solvable via an outlined technique, but the paper doesn't actually prove the case using the technique - instead the paper illustrates the technique on the $m = n+2$ case. The $m = n+3$ case was proven in [14]. These cases can be found summarized in Table 2.1.

2.4 Literature Review Summary

We have compiled in Table 2.2 a summary of known results on t-optimal graphs. In Table 2.1 we have compiled a summary of known results on t-optimal $(n.n + k)$ graphs.

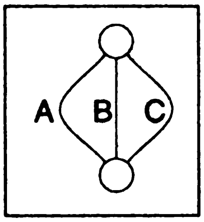
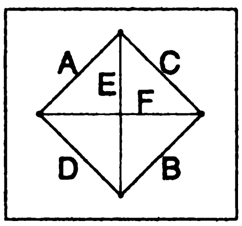
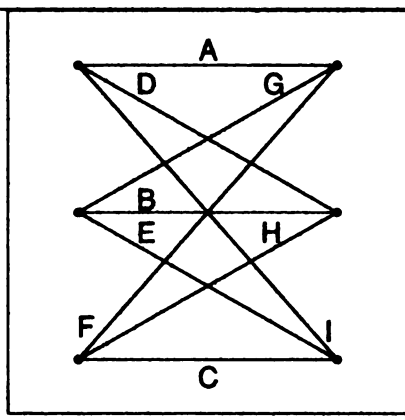
k	distillation	$\lvert chains \rvert$	equation
<-1	N/A	N/A	0
-1	any tree	N/A	1
0	a simple cycle	N/A	m
1	 This is known as a θ graph.	3	**0** $(\frac{m}{3})^3$ **1** $(\frac{m-1}{3})^2(\frac{m+2}{3})$ **2** $(\frac{m+1}{3})^2(\frac{m-2}{3})$
2		6	**0** $2\frac{m^3}{27}$ **1** $2\frac{(m-1)^2(m+2)}{27}$ **2** $2\frac{(m+1)^2(m-2)}{27}$ **3** $2\frac{m^3}{27} - \frac{m}{3}$ **4** $2\frac{(m-1)^2(m+2)}{27}$ **5** $2\frac{(m+1)^2(m-2)}{27}$
3		9	We determine this - see Table 4.1.
4	Previously unknown	N/A	Previously unknown
5	Previously unknown	N/A	Previously unknown

Table 2.1: Summary of some values of $k = m - n$ for t-optimal (n, m) graphs and their current status as of this writing.

$0 \leq m \leq n-2$: For t-optimal graphs, $t(G) = 0$
$n-1 \leq m \leq n-1$: For t-optimal graphs, $t(G) = 1$
$n \leq m \leq n$: For t-optimal graphs, $t(G) = n$
$n+1 \leq m \leq n+1$: The structure of t-optimal graphs is known. Formula is known [12] and derivable, but is piecewise by the remainder upon dividing m by 3
$n+2 \leq m \leq n+2$: The structure of t-optimal graphs is known. Formula is known [13], but is piecewise by the remainder upon dividing m by 6
$n+3 \leq m \leq n+3$: The structure of t-optimal graphs is known [14]. Formula is piecewise by the remainder upon dividing m by 9. See Table 4.1.
$n+4 \leq m \leq n+4$: The structure of t-optimal graphs previously unknown or unpublished except for when $n \leq 5$
$n+5 \leq m \leq \frac{n(n-1)}{2} - \frac{3n}{2}$: The structure of t-optimal graphs is unknown except when G is a complete regular [20] or almost regular [8] multi-partite graph or certain other cases. A formula for $t(G)$ is known for some cases, but not in terms of m and n, except for certain cases such as bipartite graphs.
$\frac{n(n-1)}{2} - \frac{3n}{2} \leq m \leq \frac{n(n-1)}{2}$: The structure of t-optimal graphs is known, with explicit closed formulas for the extremely dense cases being derivable in some cases.

Table 2.2: Summary of ranges and and their current status as of this writing

Chapter 3

Preliminary Observations And Findings

In this chapter we describe some of the preliminary research that had been done prior to formally proposing pursuing this avenue of research. Section 3.1 explains some observations upon considering some of the already existing conjectures. Section 3.2 describes some of the experimental data we had obtained as of the original proposal of pursuit of this avenue of research. Reasons to further consider these conjectures can be found in Section 3.3. The previous sections are utilized to make some new conjectures in Section 3.4. Section 3.5 describes an observation on the average number of spanning trees when we contract or delete an edge.

3.1 Previous Conjectures Revisited

In this section we consider the implications of some of the conjectures mentioned in Section 2.1. Specifically, Conjecture 1 and Conjecture 3 are used to determine the properties of the degree sequences of t-optimal graphs if those conjectures hold.

Conjecture 1 states that all graphs with maximum number of spanning trees are almost regular. Let us explore the consequences of Conjecture 1 on the degree se-

quence of a (n, m) almost regular graph G. If G is almost regular then there are k vertices with degree r and $n-k$ vertices with degree $r+1$. Since the sum of the degrees of the vertices is twice the number of edges, we have

$$2m = kr + (n-k)(r+1) = kr + nr - kr + n - k = nr + n - k = n(r+1) - k$$

From this and the fact that $k \leq n$ we can see that integers k and r are uniquely determined by m and n. If $k = n$ this is an r-regular graph and if $k = 0$ this is an $r+1$ regular graph. In all other cases, $0 < k < n$. That is, $r+1$ is the ceiling upon dividing $2m$ by n and k is $n(r+1) - 2m$. The degree sequence is thus completely determined by m and n.

For example, if we consider only graphs in the range

$$n \leq m < \frac{3n}{2}$$

then, we are considering graphs for which the following inequality holds:

$$n \leq \frac{n(r+1)-k}{2} < \frac{3n}{2}.$$

With some algebra we can see that

$$2 \leq (r+1) - \frac{k}{n} < 3$$

and thus that

$$1 + \frac{k}{n} \leq r < 2 + \frac{k}{n}.$$

We now have two cases to consider. If this graph is regular, $k = n$ or $k = 0$, and thus the graph is r regular or $r+1$ regular, respectively. If this graph is not regular, since r is an integer and $0 < k < n$ we can see that $2 \leq r < 3$, and thus $r = 2$. For specific

values of m and n the degree sequence is completely specified. If Conjecture 1 is true then t-optimal graphs where m is in the range $n < m < \frac{3n}{2}$ have maximum degree 3 and have minimum degree 2.

Let s be the number of vertices of degree 3 and k be the number of vertices of degree 2 in a t-optimal graph where m is in the range $n \leq m < \frac{3n}{2}$. Thus Conjecture 1 means that

$$2m = 2k + 3s = 2(n - s) + 3s = 2n + s$$

and thus that $s = 2(m - n)$ from which we can see how we can determine what the number of vertices of degree 3 would be given m and n. We can also see what the number of vertices of degree 2 would be.

Conjecture 3 states that among graphs satistfying $n < m < \frac{3n}{2}$, all graphs with the same number of vertices of degree 3 have the same distillation. Conjecture 3 affects the structure of this graph. Conjecture 3 means that once we know what the distillation of any t-optimal graph with s being the number of vertices of degree 3, we need only determine what the chain lengths must be for any other t-optimal graph with that same value of s, if m is in the range $n < m < \frac{3n}{2}$.

The conjectures mentioned in this section are provided purely to inform us of the direction we might take in our investigation of this problem. Their correctness is in no way assumed nor proven in the results we have obtained as of this writing.

3.2 Experimental Data

Using a piece of software called *nauty* ([1][1]) we have conducted a brute force analysis of all simple graphs with $n \leq 12$ to find the maximum number of spanning trees. (We conducted this analysis via increasing order by number of vertices, and within

[1]The use of *nauty* was very beneficial - by considering just unlabeled graphs instead of labeled graphs, we were able to do analysis on a much larger set of graphs than we otherwise would have been able to. *nauty* is the fastest known software for generating unlabeled simple graphs.

each vertex case, with increasing order by number of edges.) This data can be found summarized in Table B.1 and in Table B.2.

The analysis of the t-optimal simple $(n, n+4)$ graphs for successive values of $8 \leq n \leq 13$, has been conducted over the span of years and the $(n, n+4)$ results are found in Figures A.6, A.7, A.8, A.9, A.10, and A.11.

An artifact of the technique whereby we are determining the maximum number of spanning trees for a given n, m pair is that we are finding the number of spanning trees of all non-isomorphic simple (n, m) graphs. Since we were especially concerned with the sparsest unsolved case $(n, n+4)$ prior to our work, we present histogram data for that case, for values of n in the range $8 \leq n \leq 11$, in figures A.12, A.13, A.14, and A.15 . Figure A.12, Figure A.13, Figure A.14, and Figure A.15 are some histograms that we've constructed that show, for (n, m) graphs, what the number of graphs are that have a particular number of spanning trees. For those figures, the x axis is the number of spanning trees. The y axis is the number of non-isomorphic graphs with that number of spanning trees. One trend that can be observed is that for $(n, n+4)$ graphs, as n increases, the result gets sharper - that is, it increasingly becomes the case that the greater the number of spanning trees, the less the number of non-isomorphic graphs that has that number of spanning trees.

We have noticed that amongst all graphs encountered so far, the t-optimal (n, m) graphs are always unique up to isomorphism except in a few isolated cases. Figures A.16(a), A.16(b), A.17(a), A.17(b), A.18(a), and A.18(b) illustrate these cases. Note that Figure A.16(b) is not isomorphic to Figure A.16(a) because the vertices of degree 5 are in a cycle, Figure A.17(b) is not isomorphic to Figure A.17(a) because the vertices of degree 5 are in a cycle, and Figure A.18(b) is not isomorphic to Figure A.18(a) because there is no vertex of degree 3 that is incident to only two vertices of degree 4. Since finding those three paris of non-isomorphic t-optimal graphs, we have found some more such pairs. See section 4.5 for those.

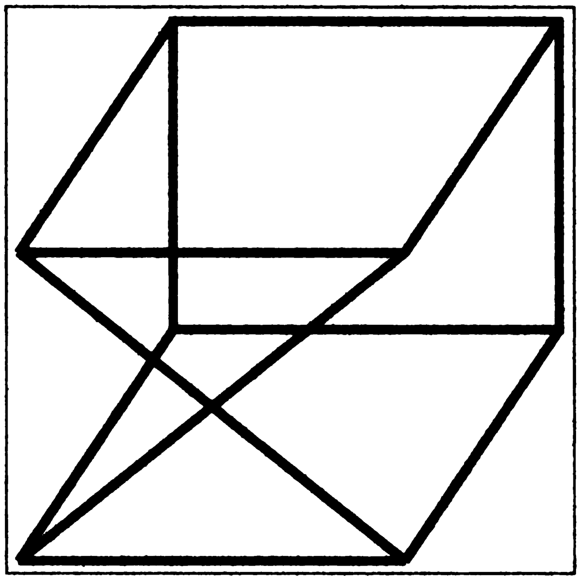


Figure 3.1: A cube with two parallel edges swapped. This graph is isomorphic to Figure A.6.

3.3 Motivation for Pursuing Previous Conjectures

With the advent of modern computers as well as a piece of software called *nauty* [1] we were able to do *brute force* analyses of all graphs with 8 vertices and 12 edges to ascertain which graphs are t-optimal. We found only one t-optimal $(8, 12)$ graph and that graph is shown in Figure 3.1 and is known as the twisted 3-cube [22]. That graph is three regular. If Conjecture 1 and Conjecture 3 hold true, then we claim this graph is the distillation of all $t(n, n + 4)$ graphs with $n \geq 8$. Note that $n \geq 8$ forces m in the range $n < m \leq \frac{3n}{2}$ for $t(n, n + 4)$. The reasoning found in Section 3.1 applies when m is in that range and we thus (by Conjecture 1) have $2(m - n)$ number of vertices of degree 3, which is 8, and all other vertices are of degree 2. Thus $t(n, n+4)$ graphs with $n \geq 8$ have the same number of vertices of degree 3 and so (if Conjecture 1 applies) Conjecture 3 applies.

The brute force analysis of the next several graphs that have 4 more edges than vertices has been conducted and so far all $(n, n + 4)$ graphs that have been analyzed (up through 13 vertices) abide by the conjectures mentioned in [15]. They can be

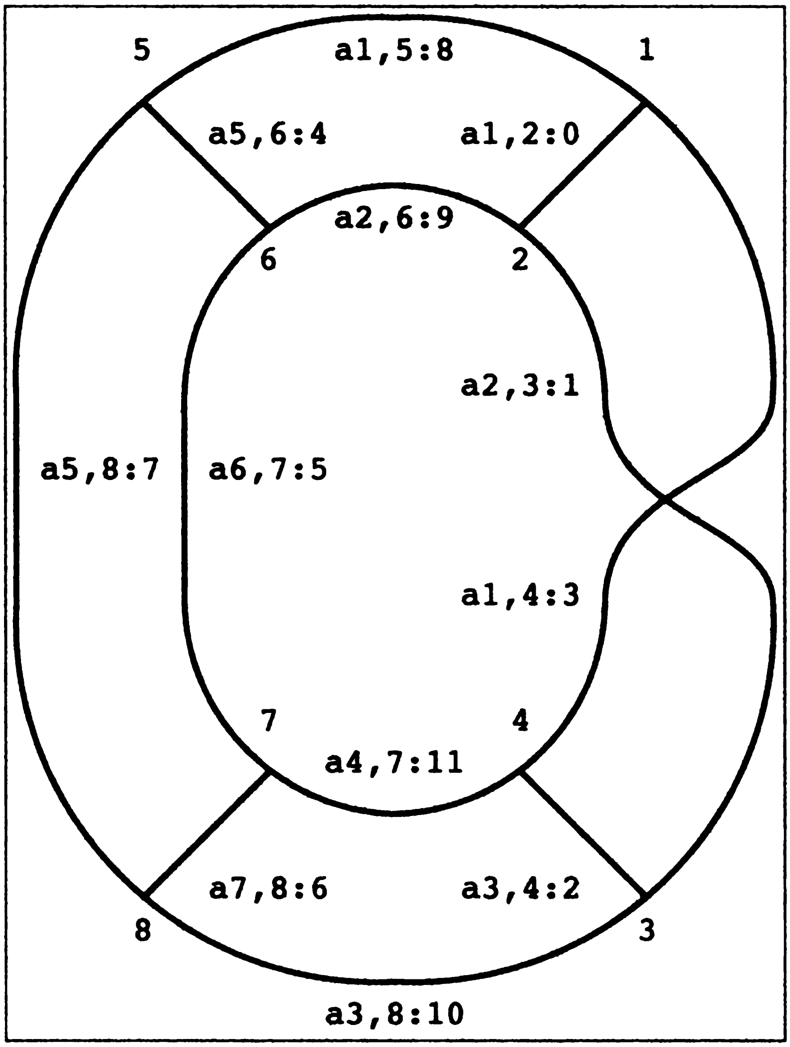


Figure 3.2: An embedding of the t-optimal graph with 8 vertices and 12 edges. The edges are labeled in the form $av_1, v_2 : e$, where v_1 and v_2 are the numbers corresponding to the adjacent vertices and e is the number associated with that edge.

found in figures A.6, A.7, A.8, A.9, A.10, and A.11.

As you can tell, inferring characteristics of the graphs given in figures A.6, A.7, A.8, A.9, A.10, and A.11 when using those embeddings is not exactly straightforward. Fortunately, we can embed the graph found in Figure 3.1 as in Figure 3.2. This embedding has a few advantages. The first advantage is that the graph has a small crossing number. Another advantage that will be important to us later is that we can see that the edges form two different classes : "spokes" and "cycle-edges". The advantage which is of the greatest concern to us right now is that for the cases with the

number of vertices ranging from 8 through 13, we can easily see the same underlying structure. Here's how: if we represent the length of the chains (where chains are as described in Section 1.2), as given in Figure 3.2 as a twelve tuple (one chain per edge in the distillation) then the t-optimal graphs for those cases can be found in Table 3.1.

(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1) corresponds to the chain lengths of Figure A.6.
(2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1) corresponds to the chain lengths of Figure A.7.
(2, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1) corresponds to the chain lengths of Figure A.8.
(2, 1, 2, 1, 1, 1, 2, 1, 1, 1, 1, 1) corresponds to the chain lengths of Figure A.9.
(2, 1, 2, 1, 2, 1, 2, 1, 1, 1, 1, 1) corresponds to the chain lengths of Figure A.10.
(2, 2, 2, 1, 2, 1, 2, 1, 1, 1, 1, 1) corresponds to the chain lengths of Figure A.11.

Table 3.1: Tuples of chain lengths for t-optimal graphs with distillation and labeling as found in Figure 3.2. Each position in the 12-tuple is the length of a chain that corresponds to a labelled edge in Figure 3.2 as per ((a1,2:0),(a2,3:1),(a3,4:2),(a4,1:3),...). The sum of the entries in the tuple is the number of edges.

This representation in terms of chain lengths also works for representing the $(n, n+3)$, $(n, n+2)$, and $(n, n+1)$ cases, which have tuples of length 9, 6, and 3, respectively. As they have periodic subdivisions, those cases can be more concisely represented via the order of the chains in which the subdivisions occur. Diagrams of those cases can be found in Table 2.1 and the order in which the edges should be subdivided are alphabetical by edge label as found in the figures in the table.

The conjectures mentioned in this section are provided purely to inform us of the direction we might take in our investigation of this problem. Their correctness is in no way assumed nor proven in the results we have obtained as of this writing.

3.4 Consequences of Conjectures

Conjectures 1, 2, and 3 hold for all graphs that have $m < n + 4$. In all such graphs, by considering each chain in turn, we may find the graph with the maximum number of spanning trees ([15] provides a summary, but the proofs and some formulas for the number of trees are actually found in [12],[13], [9] and [14]). A summary of those cases and others can be found in Table 2.1.

When we consider each chain of an (n, m) graph in turn, as can be done in the $m \leq n + 3$ cases to find the t-optimal graphs, we consider what the number of spanning trees would be if we were to increase the length of that chain by one, that is, subdividing an edge in the chain. By choosing the chain whose length, if increased by one, would give us the maximum number of spanning trees, we then have a technique that will give us the next graph with the same number of excess edges (relative to vertices). This technique will provide us with t-optimal graphs, for the $m \leq n + 3$ cases. This technique can be found summarized in Algorithm 1, and is known to be correct for $m \leq n + 3$ because it corresponds to the previously published solutions for those cases as found in [12],[13], [9] and [14].

Algorithm 1 *Pseudocode for a technique to obtain graphs with maximum number of spanning trees. Known to be correct for* $m \leq n + 3$

1. *Consider a t-optimal graph G with* $n < m \leq \frac{3n}{2}$
2. *Let* $maximumSoFar = 0$
3. *Let* $bestGraphSoFar = G$
4. *For each chain c in the distillation of G :*
   (a) *Consider G' which is G with the length of c increased by 1.*
   (b) *If* $t(G') > maximumSoFar$ *set* $maximumSoFar = t(G')$ *and set* $bestGraphSoFar = G'$

5. *bestGraphSoFar now contains the next graph that this algorithm produces*

Lemma 1 *Algorithm 1, for all graphs with $m \leq n+3$, has a periodic pattern consisting of all of the chains. For each chain in the sequence, we increase the length of the chain, and go on to the next chain in the sequence. (Proved in [12],[13], [9] and [14])* $\square$

Lemma 1 is true for all graphs with $m \leq n+3$. This gives us reason to conjecture that it is true whenever started from a t-optimal graph for which $m \leq \frac{3n}{2}$, and thus we make the following conjecture:

Conjecture 4 *Algorithm 1 always generates graphs with maximum number of spanning trees.*

On a per distillation basis, a formula for $t(G)$ in terms of the chain lengths of G can be found by considering that for every tree in a graph, there is a corresponding tree in the distillation of the graph. If all of the edges in a chain c are in a spanning tree T of the graph G then the edge e in the distillation $D(G)$ that corresponds to c is in the corresponding (to T) spanning tree in the distillation $D(G)$. Likewise, if any edge in the chain is not in that spanning tree of the graph (there can be only one such edge per chain or else the "spanning tree" would be disconnected and hence not a spanning tree), then the edge in the distillation that corresponds to that chain is not in the spanning tree in the distillation. Thus, the formula for $t(G)$ has a number of terms equal to the number of trees in the distillation, and each term is the product of the chain lengths of the chains that correspond to edges not in the tree in the distillation.

Using Algorithm 1 results in a linear operation to find the $(n, n+4)$ graph (because the number of chains has a constant upper bound, namely 12 in this case). Without the realization that subdividing any edge in the chain increases the size of the chain, we would need to consider every edge instead of every chain, and thus, since the number of edges (m) grows, finding a t-optimal graph would be order m^2, a polynomial time operation. There are certain conditions necessary to ensure that this realization works for finding t-optimal graphs. Those conditions are that all $(n, n+k)$ graphs have the same distillation and are such that for any n, the chain length tuples for n and $n+1$ differ in only one chain. Those conditions are true for the simpler cases ($m \leq n+3$).

When we consider Lemma 1, we are naturally led to consider what happens for $(n, n+4)$ graphs when Algorithm 1 is applied. We find the sequence of chains is non-periodic.

Consider the set of permutations that are graph automorphisms. We say two edges are in the same class if and only if there exists some graph automorphism permutation that maps from one edge to the other. Let $m_{i,G}(e)$ denote the edge that edge e of graph G is mapped onto using automorphism i. Let $M_G(e)$ denote the set of $m_{i,G}(e)$ for all possible i that map edges on to one another. Including the identity permutation, $M_G(e)$ defines a class of edges to which e belongs, and the application of M_G to any edge in a class will result in the same class.

If we apply this way of finding classes of edges to the distillation of a graph, because each edge in the distillation corresponds to a chain, we then find classes of chains in the graph. Let $d_G(c)$ represent the edge in the distillation $D(G)$ of G that the chain c of G corresponds to. Thus, we can find classes of chains in the graph G by considering the classes of edges of $D(G)$ and then applying the inverse of $d_G(c)$ to each element of these classes. In all the simpler cases ($m \leq n+3$), all edges in the distillation were part of the same class. But in the $(n, n+4)$ case there are two classes instead of one. These are the classes of chains mentioned in Section 3.3.

Knowing that we have two classes of chains, we can investigate how they should relate to each other. Knowing that all the graphs ($m \leq n+3$) have chains being about the same in length, we make the following conjecture (Conjecture 5):

Conjecture 5 *In a t-optimal graph, all chains in the same class will have lengths that differ by no more than 1.*

We have discovered via brute force that $t(8, 12)$ has 2 and only 2 classes of chains. By using the formula for the number of spanning trees in terms of the chain lengths, which is unique for each distillation, and the fact that the sum of the chain lengths is the number of edges, and Conjecture 5, we created a system of equations, two equations, two unknowns, and so we solved and optimized for the number of spanning trees. When we optimized for the number of spanning trees given the equations, we found that the optimal ratio of the number of edges in these different classes is an irrational number, namely $\sqrt{5} - \sqrt{2}$. This finding would explain why we do not have the same kind of periodically repeating ordered progression as found in the $m \leq n+3$ cases.

The set $S(10, 15)$ contains one graph which is the Petersen graph. It has one class and thus, by Conjecture 5, ought to have a periodic progression when we apply Algorithm 1. When we apply the algorithm, we get the results found in Table B.3, expressed in terms of chain lengths. From this we can see that we have such a periodic progression (not shown in the table for the sake of brevity, but the progression does repeat periodically). The periodic progression is the following:

$$1, 9, 15, 2, 10, 12, 3, 6, 14, 4, 7, 11, 5, 8, 13$$

The labeling of the edges of the Petersen graph is as found in Figure A.5 and follows the same convention as in Figure 3.2.

In summary, we conjecture that chains in the same class have lengths that differ by no more than one (Conjecture 5). We also conjecture that the technique described in Algorithm 1 always gives a t-optimal graph if started from a t-optimal graph with maximum degree of 3 (Conjecture 4). These conjectures are not necessary for the results we obtain.

3.5 The Average Number of Spanning Trees upon Deletion Or Contraction

If we consider a graph and construct a matrix with every spanning tree of the graph, one per column, and every edge, one per row, then we can count the edges in each tree by summing up each column separately and we get $(n-1)t(G)$. If we sum up each row separately we get the sum over every edge of the number of spanning trees containing this edge. If we use $t(G\cdot\bar{e})$ to represent the average number of trees upon contracting an edge e, then we get the equality

$$(n-1)t(G) = (m)t(G\cdot\bar{e}).$$

Similarly if we count the number of edges that aren't in the tree and we use $t(G-\bar{e})$ to represent the average value thereof, we get

$$(m-(n-1))t(G) = (m)t(G-\bar{e}).$$

These two equations can both be rewritten as

$$t(G) = \frac{(m)t(G\cdot\bar{e})}{n-1}$$

and

$$t(G) = \frac{(m)t(G-\bar{e})}{m-(n-1)}.$$

Putting these equations together, we get

$$\frac{t(G \cdot \overline{e})}{t(G - \overline{e})} = \frac{n-1}{m-(n-1)}.$$

This equation can be useful when considering the Feussner equation mentioned in Section 1.2 [11]. Note that there may be no graph that has $t(G - \overline{e})$ or $t(G \cdot \overline{e})$ spanning trees, but as an average we know there is at least one graph with number of spanning trees greater than or equal to it and that there is also at least one graph with number of spanning trees less than or equal to it. By applying these equations we may be able to construct bounds on extremal values of $t(G)$.

Chapter 4

Further Results

With the advent of modern computers as well as a piece of software called *nauty* [1] we were able to do analyses of a large number of graphs. Some of the data we accumulated from that can be found in [23]. The data and analyses thereof help to inform some of the following few sections. In Section 4.1 we prove an important lemma concerning distillations. We share some observations concerning classes of chains in Section 4.2. In Section 4.3 we describe our technique for proving the formula for $t(n, n+3)$. We present, with proof, our results for the $(n, n+4)$ case, in Section 4.4. The more recent results of our empirical analysis are presented in Section 4.5. In Section 4.6 we begin to lay a possible groundwork for being easily able to extend some of our results beyond the $(n, n+4)$ case.

We might wonder if the removal of an edge from a t-optimal (n, m) graph will always produce a t-optimal graph. We do not always get a t-optimal graph under such circumstances as it depends on which graph we are removing the edge from. For example, any t-optimal θ graph (a θ graph is a biconnected graph with two vertices of degree three and all other vertices of degree two) for which $n \geq 5$ has the property that removing a single edge will result in a graph with a vertex of degree one. However, graphs with a vertex of degree one cannot be t-optimal because t-optimal graphs are biconnected [9]. On the other hand, complete graphs are t-optimal and removing any single edge from one of them results in a t-optimal graph.

Similarly, we might wonder if the addition of an edge to a t-optimal graph always results in a t-optimal graph. Clearly we would want to add the edge in the right "spot". We might also wonder if there is always a "right spot" to add the edge. There is not always such a "right spot". Again, if we have a cycle, which is a t-optimal graph, and we add an edge connecting any two non-adjacent vertices, we get a θ graph, but if the cycle is larger than size 4, the resulting graph will not be t-optimal. However, sometimes adding any non-parallel edge will give us a t-optimal graph, as in the case of $(n, \binom{n}{2} - 1)$ graphs.

4.1 A Lemma Concerning Distillations

The concept of the distillation, described earlier and in [9] is useful to us. We can categorize all biconnected graphs of interest to us in terms of their distillations. The following lemma helps us do that, and from there we are able to explore the consequences of a graph having a particular distillation.

Lemma 2 *For any biconnected $(n, n+k)$ graph there is a corresponding $(2k, 3k)$ graph with the same distillation.*

Proof: A biconnected graph has no vertices of degree zero or one. The distillation process results in a graph with no vertices of degree 2 by definition. Thus, distillations of biconnected graphs have a minimum degree of 3. We prove Lemma 2 by contradiction and thus assume $n \neq 2k$ and consider two cases.

Case 1 : Assume $n > 2k$, and consider biconnected $(n, n+k)$ graphs with more than $2k$ vertices. Suppose $h = n - 2k$. Thus we should consider biconnected $(2k + h, 3k + h)$ graphs. Of these biconnected graphs, there must be at least

h vertices of degree 2. Suppose not, and we only have $h-1$ vertices of degree 2. Since all the other vertices are of degree 3 or greater we would have at least $3k+h+\frac{1}{2}$ edges, which is greater than $3k+h$. Thus there are at least h vertices of degree 2. Thus we can contract an edge incident to each of them and get a $(2k, 3k)$ graph with the same distillation.

Case 2 : Assume $n < 2k$, and consider biconnected $(n, n+k)$ graphs with less than $2k$ vertices. Suppose $h = 2k - n$. Thus we consider biconnected $(2k-h, 3k-h)$ graphs. If we consider an edge and subdivide it, we get a $(2k-h+1, 3k-h+1)$ graph. If we repeat this h times, we get a $(2k, 3k)$ graph with that same distillation.

So, in either case, for any $(n, n+k)$ graph there is a $(2k, 3k)$ graph with the same distillation. □

4.2 Chain Classes

We have found the techniques that were used to prove the optimal solutions for the $(n, n+k)$ cases when $k \leq 3$ to be difficult to apply to the $(n, n+4)$ case. We believe this difficulty is due to an important difference between the $(n, n+4)$ case and the other cases; this difference can be seen by considering equivalence classes under isomorphic mappings of edges on the edges in the distillation, and thus the chains in the undistilled graph (as described in Section 3.4).

By drawing the graph found in Figure 3.1 as in Figure 3.2 we can see that the edges form two different classes : "spokes" and "cycle-edges" as mentioned in Section 3.3. For the cases with the number of vertices ranging from 8 through 13 (that is, Figures A.6, A.7, A.8, A.9, A.10, and A.11), we easily see the same distillation by representing the length of the chains, as given in Figure 3.2, as a twelve tuple (one

chain per edge in the distillation). Thus the t-optimal graphs for those cases are found in Table 3.1.

For an $(n, n+k)$ graph G, if $k > \frac{n}{2}$ then by necessity G will be forced to have a vertex of degree greater than 3. Algorithm 1 will not be able to remedy this if the optimal distillation has vertices of degree 3 only. Thus, to ensure that Algorithm 1 is started from a graph that could have vertices of degree 3 only, we only start it when $k \leq \frac{n}{2}$.

If we start with the graph found in Figure 3.2 and iterate Algorithm 1 a few times, then Table 3.1 contains the chain length tuple values. Between each tuple, the changed element is the one whose value is increased and so is the element whose increase was what was maximum.

For a particular distillation, we only need to concern ourselves with chain length value assignments in the attempt to find a t-optimal graph with that distillation. This is due to the formula mentioned in Section 3.4. Using Algorithm 1 to determine these chain length value assignments results in a linear operation to find the $(n, n+k)$ graph (because the number of chains has an upper bound of $3k$ when $m \leq \frac{3n}{2}$ which follows from Lemma 2).

As mentioned: we conjecture that chains in the same class have lengths that differ by no more than one (Conjecture 5). Everything mentioned in this section would support, but not prove, that conjecture.

4.3 The Formula for $t(n, n+3)$

Given the formula mentioned at the beginning of Section 3.4 and a repeating ordered progression of all chains in a graph we can see that increasing the lengths of the chains in accordance with the repeating ordered progression will result in polynomial expressions for each remainder upon division of the number of edges by the length of the period. Furthermore, that formula also enables us to see that the maximum degree of a term in these polynomial expressions is $1 + k = 1 + (m - n)$. This insight is consistent with previously published formulas for $t(n, m)$ in the $m \leq n + 2$ cases. While the distillation for the optimal $(n, n+3)$ case has been shown in previous work [14], as far as we know, no formula has been previously published for $t(n, n + 3)$. Thus, we can use this insight to determine a formula for $t(n, n + 3)$ when $n \geq 6$.

Let r be the remainder upon dividing m by the number of chains c in an (n, m) graph. When the pattern of edge subdivision as described in Algorithm 1 periodically repeats and the length of the period is c, then for two graphs with m values m_0 and m_1, if they differ by exactly c, then they will be the same except the larger will have had an edge from every chain subdivided with respect to the smaller. For this reason, as found in the $m = n+3$ case, it is convenient to represent the formula for $t(n, n+k)$ as a piecewise function depending on the value of r.

Given the algorithm for the $m = n + 3$ case [14], and the information in the previous paragraph, we now have the formula for $t(n, n + 3)$ (obtained by iterating Algorithm 1, looking at the result, and determining the polynomial from the result), as found in Table 4.1.

4.4 Theoretical Results

In this section we utilize our observations in terms of chain length tuples and per distillation expressions. We determine that the expression for the number of spanning trees in terms of the chain lengths has an optimization solution equation that is a polynomial whose local maximum is also the global maximum. We also determine that the optimal integer valued solution will be within decreasing bounds of the optimal continuous valued solution which is expressible in terms of per distillation fixed chain length ratios. Altogether, this enables us to state that Algorithm 1, applied to the

Number of spanning trees	remainder upon division of $n-6$ by 9
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{2n^2}{3}+\frac{4n}{3}+1$	0
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{16n^2}{27}+\frac{80n}{81}+\frac{16}{27}$	1
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{5n^2}{9}+\frac{58n}{81}+\frac{8}{27}$	2
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{5n^2}{9}+\frac{2n}{3}$	3
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{14n^2}{27}+\frac{38n}{81}-\left(\frac{4}{27}\right)$	4
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{14n^2}{27}+\frac{52n}{81}+\frac{7}{27}$	5
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{14n^2}{27}+\frac{4n}{9}$	6
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{5n^2}{9}+\frac{50n}{81}$	7
$\frac{n^4}{81}+\frac{4n^3}{27}+\frac{16n^2}{27}+\frac{64n}{81}$	8

Table 4.1: The piecewise formula for the number of spanning trees of t-optimal $(n, n+3)$ graphs. The left column is the formula, and the right is the remainder upon division of $n-6$ by 9.

correct distillation, will always result in an optimal assignment of values to chain lengths eventually.

There are some properties of optimal chain length value assignments that we would like to know. A chain length value assignment local maximum is a global maximum; this is important because it helps us determine some of those properties.

Lemma 3 *Consider the set of all vectors V of length γ, the sum of whose elements is a constant τ, that is,*

$$V = \{v_i | v_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,\gamma}) \wedge \sum_{j=1}^{\gamma} x_{i,j} = \tau\}$$

where $x_{i,j} \in \mathbb{R}$.

Under such conditions, V is a convex set.

Proof: The sum of the average is the average of the sum, because the denominators in computing the average (the length of the vector, which is fixed) are all the same, and because addition is commutative and division is left distributive over addition. Therefore the average of the vectors will have the same sum as the vectors themselves have and thus will be in the same set. Thus the definition of convex set applies. □

Lemma 4 *Consider the set of all vectors V of length γ, the sum of whose elements is a constant τ, that is,*

$$V = \{v_i | v_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,\gamma}) \wedge \sum_{j=1}^{\gamma} x_{i,j} = \tau\}$$

where $x_{i,j} \in \mathbb{R}$.

Let us further add the restriction that $x_{i,j} > 0$.

Consider the set of functions F on vectors $v \in V$ such that

$$F = \{f | f(x_1, x_2, \ldots, x_\gamma) = \sum_{j \in P([\gamma])} a_{f,j} \Pi_{x \in j} x\}$$

where $a_{f,j} \in \{0,1\}$*;* $[\gamma]$ *is the set of all the* x_i *for* $1 \leq i \leq \gamma$*; and* $P(s)$ *is the power set of set* s*.*

Under these situations, every such function $f \in F$ *is a concave function.*

Proof: Each term in the sum is greater than or equal to the average of the corresponding terms as shown by a proof by induction on the length of the vector:

1 They are equal because they are the same.

2 $f(\frac{(x_1,y_1)+(x_2,y_2)}{2}) \geq \frac{f(x_1,y_1)+f(x_2,y_2)}{2}$

$\iff$

$\frac{(x_1+x_2)(y_1+y_2)}{2} \geq \frac{x_1y_1+x_2y_2}{2}$

$\iff$

$\frac{x_1y_1+x_2y_1+x_1y_2+x_2y_2}{2} \geq \frac{x_1y_1+x_2y_2}{2}$

$\iff$

$\frac{x_2y_1+x_1y_2}{2} \geq 0$

And thus we are done since x_1, y_1, x_2, y_2 are all ≥ 0

Inductive step The induction hypothesis is true for n elements in the vector.

Now suppose we have $n+1$ elements and j is element $n+1$. Factor the elements in a manner similar to how the base case 2 step was done. Observe that we will have all the terms on the right on the left, as well as terms on the left involving the product of a j with elements that were not in the same vector as that j.

Thus Lemma 4 is proven. □

A concave function on a convex set has the property that the local maximum is global maximum. That fact is useful to is. Lemma 3 and Lemma 4, together mean that we can apply that fact to the problem of determining values for chain lengths,

Let $G_{d(g)}(u)$ be the graph with distillation g and chain length tuple u. Imagine, for a given distillation g, we have a contour plot of $t(G_{d(g)}(u))$, where the dimensions are the chain lengths and with contour lines drawn at the integer values (that is a way to visualize this with only two variables). If we look at the subspace defined by any straight line, we see that the values on the contour plot that the line intersects form a parabola due to the fact that we have two variables, let us call them x and y and, since the line defines values for all the variables other than x and y and the sum of the variables is constant (that is the restriction that makes this a convex set), we have $a + bx + cy + dxy$ over $t = x + y$, where t is the constant (after the values for all the other variables have been fixed to determine a, b, c, and d). So if we express $a + bx + cy + dxy$ in terms of x and t (with no y) then we have $a+bx+c(t-x)+dx(t-x) = a+bx+ct-cx+dtx-dx^2 = (a+ct)+(b+dt-c)x-dx^2$. Clearly this is, expressed in terms of x, a parabola, with the local maximum being the global maximum. This should give us a more intuitive understanding of the consequences of, and rationale for, understanding that V (as mentioned in Lemma 3) is a convex set.

Lemma 5 *For a graph with a specific distillation, the optimal assignment of continuous values to chain lengths is expressible in terms of fixed ratios between chain length values.*

Before we prove Lemma 5, let us consider an example. The θ graph is a graph with 3 chains. Consider the (n, m) graph found in Figure 4.1. Let a, b, and c represent the lengths of chains A, B, and C, respectively. One way of getting a spanning tree would be to remove edge uv from chain A and edge wx from chain C. That way of getting a spanning tree is just a special case of removing any edge from chain A and any

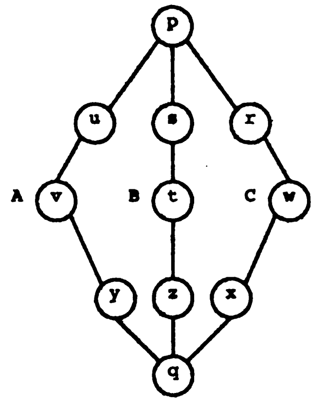


Figure 4.1: A θ graph

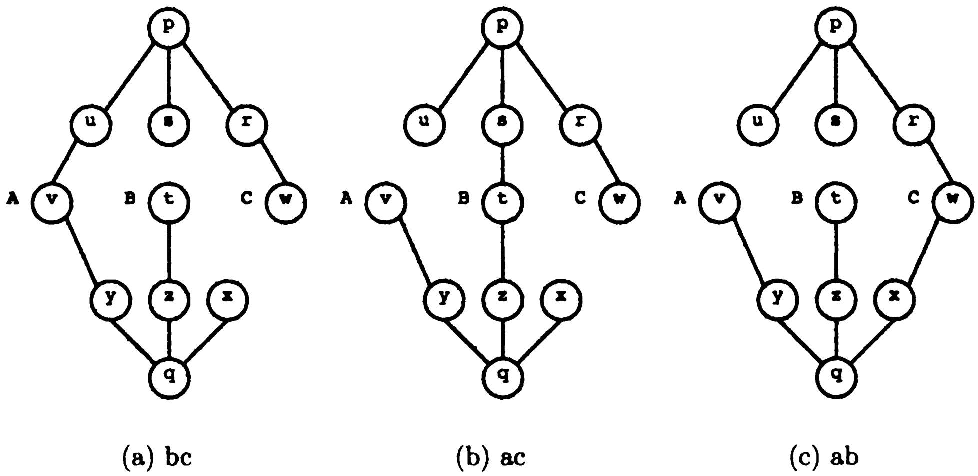


Figure 4.2: In a θ graph, possible ways to have two chains with one edge removed.

edge from chain C. This can be found depicted in Figure 4.2(b). There are ac such possible spanning trees. Similarly, for chain pairs B, C and A, B this yields bc and ab spanning trees as well, respectively, as depicted in Figures 4.2(a) and 4.2(c). For each set of chains which must have edges removed, the number of possible spanning trees is the product of the chain lengths, and thus we add up for each set and get

$$ab + bc + ac$$

which we want to maximize subject to $a + b + c = m$. Thus, clearly in this case, the optimal continuous solution ratio of chain length values to m is $\frac{1}{3}$.

Proof: First observe that, since the expression for the number of spanning trees in terms of the chain lengths is a polynomial, that which uniquely optimizes the leading coefficient optimizes the solution.

We induct on the number of variables (which is the number of chains).

Base cases When there is one chain the ratio between the chain length and the number of edges will be a ratio of 1, by definition. See the argument given in the "more intuitive understanding" discussion after Lemma 3 regarding the two-dimensional case being a parabola.

Inductive step Assume for n, prove for $n + 1$

Amongst the $n + 1$ chains, take an arbitrary chain (call it k) and fix the length of it. Applying the induction hypothesis, solve for the other n chains. The optimal value for that is thus expressed in terms of k and the total t. The optimal value of k can be found symbolically because of the nature of the expression: the expression is $f_0(t - k) + f_1(t - k)k$, f_0 and f_1 are increasing polynomial functions; let the degree of the leading term of f_0 be d, and d is one greater than the degree of the leading term of f_1. Let a be the leading

coefficient of f_0 and b be the leading coefficient of f_1. Bear in mind that a and b are independent of t and k.

Now we determine the k that maximizes $a(t-k)^d + b(t-k)^{d-1}k$.

$a(t-k)^d + b(t-k)^{d-1}k = (t-k)^{d-1}(a(t-k) + bk)$

$\Longleftrightarrow$

$a(t-k) + bk = at + (b-a)k$

So we take $(t-k)^{d-1}(at + (b-a)k)$ and differentiate it with respect to k and set it equal to zero and solve for k in terms of t. The solution is

$k = t\frac{ad-b}{ad-bd}$

which clearly indicates that the value of k has a fixed ratio with respect to t, as d is fixed and a and b are the values of fixed ratios with respect to t due to the induction hypothesis.

Thus, since we have proven it for the inductive step, the induction proof is complete and we have proven Lemma 5. □

Lemma 6 *Let x_m be the length of a chain in an (n, m) graph produced by Algorithm 1. Let x_{m+1} be the length of the same chain when the number of edges is $m+1$, that is, after one iteration of Algorithm 1. Then*

$$|\frac{x_m}{m} - \frac{x_{m+1}}{m+1}| \leq \frac{1}{m}$$

Proof: When applying a step in Algorithm 1, x_{m+1} is either x_m or $x_m + 1$ and thus respectively either the difference is $\frac{x_m - 1}{m(m+1)}$, which is clearly less than $\frac{1}{m}$ for all positive m because x_m is less than m, or the difference is $\frac{x_m}{m(m+1)} - \frac{1}{m+1}$ which is less than $\frac{1}{m}$ because x_m is less than m and an integer strictly greater than zero. □

Let the chain length tuple α be given by Algorithm 1, starting with the distillation, and let chain length tuple β be a least integer unique optimal solution such that for all elements in α, the corresponding element in β is greater or equal.

Define $dis(\alpha, \beta)$ as $\sqrt{\sum_{i \in |(|\alpha|=|\beta|)|} (\beta_i - \alpha_i)^2}$

Lemma 7 *Given a chain length tuple α_0 such that α is α_0 with one iteration of Algorithm 1 applied to it, Algorithm 1 will choose an edge that minimizes $dis(\alpha, \beta)$.*

Proof: Since all possible edge choices are equidistant in the chain length tuple space, and Algorithm 1 chooses the one with the greatest number of spanning trees, and β is optimized, it chooses the one that minimizes $dis(\alpha, \beta)$ (if not, then the solution space is not convex). □

Lemma 8 *The edge that minimizes $dis(\alpha, \beta)$ is the edge that, after the subdivision causes $(\beta_i - \alpha_i)^2$ to have the greatest value.*

Proof: Any other edge, besides the one that minimizes $dis(\alpha, \beta)$ won't reduce $dis(\alpha, \beta)$ as much as the one that has the greatest $(\beta_i - \alpha_i)^2$ value. □

Lemma 9 *Let the chain length tuple α be given by Algorithm 1, starting with the distillation, and let chain length tuple β be a least integer unique optimal solution such that for all elements in α, the corresponding element in β is greater or equal. Applying Algorithm 1 to chain length tuple α will eventually yield β.*

Proof: We prove this by induction on the sum of the differences between the elements of β and α.

Base case The sum of the differences is 1. This case follows from the definition of optimal and the greedy nature of Algorithm 1.

Inductive step Since the lemma holds for n as per the induction hypothesis, that means that the induction hypothesis works for α' which is α after one algorithm

iteration has been applied to α, as long as the edge added in that iteration is in one of the chains for which the chain length in β is greater than in α.

If we see this as a multi-dimensional space, then Algorithm 1 always chooses the point that "climbs the mountain" the fastest. If we were to increase the length of some chain that wouldn't lead us to the optimal, it would have to be because of a "curve in the mountain" which cannot exist because it is a convex set (Lemma 3).

This can more formally be seen as a consequence of Lemma 7 and Lemma 8.

Thus Algorithm 1 will choose a chain for which the number of edges in β is greater than the number of edges in α.

Thus Lemma 9 is proven. □

Corollary 1 *For each chain x, the ratio $\frac{x_m}{m}$ will converge to the continuous optimal solution as $\text{limit}_{m \to \infty}$, where x_m is as defined in Lemma 6.*

Proof: The convergence is a consequence of Lemma 9 and Lemma 6. □

Lemma 10 *Zero is the optimal ratio of the length of a chain x in a graph G to the number of edges in G if and only if removal of the edge e (the edge in the distillation $D(G)$ that corresponds to the chain x in G) results in a graph that is not biconnected.*

Proof: Removal of an edge e in the distillation results in a connected graph $G-e$ that is not biconnected. Since $G-e$ is not biconnected, but G was, $G-e$ must be connected and it must have a cut vertex v. Let A be the part of $G-e$ on one side of the cut vertex (and containing one of the end vertices of e) and let B be the part of $G-e$ on the other side of the cut vertex (and containing the other end vertex of e).

$t(G) = t(A)t(B)x + t_{2,x,v}(A)t(B) + t_{2,x,v}(B)t(A)$

where $t_{2,y,w}(H)$ is the number of two-component forests of H where y is in one component and w is in another.

Without loss of generality, let $|E(A)| - |V(A)| \geq |E(B)| - |V(B)|$. Let us rewrite the above equation, but reflecting that A is a function of the number of edges that it is allowed:

$t(G) = t(A(u))t(B)x + t_{2,x,v}(A(u))t(B) + t_{2,x,v}(B)t(A(u))$

The optimal ratio having the value zero is the same as

$t(A(u+1))t(B)(x-1)+t_{2,x,v}(A(u+1))t(B)+t_{2,x,v}(B)t(A(u+1)) > t(A(u))t(B)x+$
$t_{2,x,v}(A(u))t(B) + t_{2,x,v}(B)t(A(u))$

$\iff$

$(t(A(u+1))-t(A(u)))t(B)(x-1)+t(A(u))t(B)+(t_{2,x,v}(A(u+1))-t_{2,x,v}(A(u)))t(B)+$
$t_{2,x,v}(B)(t(A(u+1)) - t(A(u))) > 0$

The latter is clearly true as $t(A(x))$ and $t_{2,x,v}(A(y))$ must be increasing functions and since $x \geq 1$ (otherwise it is zero and the proof is finished either way). □

Corollary 2 *t-optimal graphs do not have distillations in which there is an edge whose removal results in a graph that is not biconnected.* □

We will now resume a series of proofs and corollaries relating to how Algorithm 1 and the (optimal) chain length ratios interplay with one another.

As before, let the chain length tuple α be given by Algorithm 1, starting with the distillation, and let chain length tuple β be a least integer unique optimal solution such that for all elements in α, the corresponding element in β is greater or equal. Lemma 5 and Lemma 9 mean that Algorithm 1 is "self-correcting" - that is for every α there exists some β and thus after some finite number of iterations applied to any α, we will have an optimal solution β.

From Lemma 9 we have the following corollary:

Corollary 3 *If Algorithm 1 is applied to a graph whose integer chain length tuple*

value assignments are optimal then Algorithm 1 enables us to retain the property of having optimal integer chain length value assignments if there is a repeating and periodic sequence of chains that are to be subdivided. Even if there is not such sequence, the greediness of the algorithm combined with the convexity of the space means Algorithm 1 will produce a sequence of graphs whose number of spanning trees will, subject to integer restrictions, be asymptotic to a polynomial with the same leading coefficient as the not-necessarily-integer optimal solution.

Corollary 4 *After some number of iterations, for a chain c which Algorithm 1 would choose to increment in length, assuming Algorithm 1 was initially started with a distillation, prior to any series of incrementations of c, the ratio of the length of c to the overall number of edges is, within bounds of Lemma 6, a lower bound on the optimal ratio.*

Proof: Corollary 4 follows from Lemma 7 and Lemma 8. □

Similar to Corollary 4, we can infer from Lemma 7 and Lemma 8 a technique for finding an upper bound as well.

Corollary 5 *For a chain which Algorithm 1 has finished incrementing the value of its length (that is, the next iteration wouldn't increment the same chain length as well), an upper bound on the optimal ratio can be determined, within the range specified by Lemma 6.* □

By using this upper bound (from Corollary 5) in the equation, we can infer an upper bound on the leading coefficient (similarly for a lower bound as well, using Corollary 4). Thus, for a given distillation, from Corollary 4 and from Corollary 5 we can determine lower and upper bounds on the leading coefficient for an optimal assignment of values to chain lengths.

Given a particular distillation, and that the local solution is the global solution in the continuous case, we can use the Algorithm 1 sequence to determine bounds on the leading coefficient of the continuous optimal solution polynomial expression. Given those bounds, if the lower bound of one application to a particular distillation is greater than the upper bound of all the others, then that is the optimal distillation.

We ran enough iterations for the lower bound of one distillation to be greater than the upper bound of all the other possible distillations of $(n, n+4)$ graphs, excluding the cases described in Lemma 10 and Corollary 2. That one distillation is the twisted 3-cube. Thus, by applying Algorithm 1 to the twisted 3-cube, we either get the optimal solution or by Corollary 3 something very close.

The sequence of chains is not periodic when Algorithm 1 is applied to the twisted 3-cube. We endeavor to consider the consequences of this. First we will consider when the sequence of chains might be periodic, and then we will be better able to discuss why the sequence of chains is not periodic when Algorithm 1 is applied to the twisted 3-cube.

Given that we have two classes of chains, we can further investigate the conjecture that chains in the same class differ in length by no more than 1. Given that all the previous t-optimal (n, m) graphs $(m \leq n+3)$ have chains about the same length, the following lemmas make sense:

Lemma 11 *If optimal ratios of chain lengths for a particular distillation are rational values then continuous optimal chain lengths are integer values for some infinite subset of the optimal solution (usually at least one piece of a piecewise function).*

Proof: The least common multiple of the denominators of the ratios, when expressed as smallest possible fractions, will, by definition, be an integer such that, for every multiple thereof that is a number of edges, the chain lengths will have integer values that are optimal ratios. □

Lemma 12 *If optimal ratios of chain lengths for a particular distillation are rational values then there is a periodic progression of chains to subdivide.*

Proof: The least common multiple of the denominators of the ratios, when expressed as smallest possible fractions, will, by definition, be an integer that, for every multiple thereof that is a number of edges, the chain lengths will have the same ratios. Thus, the progression of chains to subdivide will be periodic with the period being that least common multiple. □

Lemma 13 *If, in a t-optimal graph, there is a periodic progression of chains to subdivide, then all chains in the same class have lengths that differ by no more than 1.*

Proof: The period will not repeat until all the chains have been incremented at least once (every chain must be represented due to Lemma 10 as Corollary 1 means that if a chain is not represented in the period then its optimal ratio would be zero which doesn't happen). Corollary 1 also means that the periodic progression will dominate the determination of the ratio over time. □

Optimal ratios being rational values, along with the fact that, as stated earlier, Lemma 3 and Lemma 4 together mean that since we can apply the fact that a concave function on a convex set has the property that the local maximum is global maximum to the problem of determining values for chain lengths, Lemma 13 holds under those circumstances.

Now that we have established that rational optimal chain length ratios in the continuous solution mean a periodic progression of chain length subdivisions, we can explore why we do not see this in the $(n, n+4)$ case. Restating from Section 3.4, by using the equation for $t(G)$ in terms of the chain lengths of G, which is unique for each distillation up to isomorphism, and the conjecture that all chain lengths in the same class are about the same length[1] and the fact that the sum of the chain lengths is the number of edges, we can create a system of equations, to be specific, two equations, two unknowns, and thus solve and optimize for $t(G)$ (as mentioned in Section 3.4).

[1]which is shown to be true for certain cases such as in Lemma 13

When we, given the equations, optimize for $t(G)$, the optimal ratio of the number of edges in these different classes is an irrational number, namely $\sqrt{5} - \sqrt{2}$. This suggests why we do not have the same kind of periodic progression of chain lengths to increase as found in the $m \leq n + 3$ cases[2]. Thus, all we can work with is Corollary 3 in the $(n, n + 4)$ case.

As mentioned in Section 3.4, we have considered the $(n, n+5)$ case and do have a conjecture as to the distillation (the Petersen graph) and orderly periodic subdivision progression. However, we do not have a proof of this as of yet via the same means as we obtained for the $(n, n + 4)$ case. Specifically, application of Corollary 4 and Corollary 5 to the $(n, n + 5)$ case is problematic because of the vastly larger number of possible distillations to consider.

4.5 Empirical Results

We used *nauty* to do analyses of as many graphs as we could to find which ones were t-optimal. Our data for the values of $t(n, m)$ can be found in Tables B.1 and B.2.

During our investigation using that technique, we determined that t-optimal graphs are not unique; see Section 3.2 for some of the earlier results we obtained regarding this. Since then we have found some additional pairs of non-isomorphic t-optimal graphs, as found in Figure A.19 and Figure A.20. Alternatively this can be viewed as considering $|S(n, n + k)|$ to determine how unique the t-optimal graphs are. Our results are listed in greatest detail in [24]. As you can see, for most $n, n+k$, $|S(n, n + k)| = 1$. Reiterating from before, the five cases encountered so far where $|S(n, n+k)| > 1$ can be found in Figure A.16, Figure A.17, Figure A.18, Figure A.19, and Figure A.20.

[2] It also means that Lemma 13 does not necessarily apply to the $(n, n+4)$ case as the subdivisions are not in a periodic pattern, one of the conditions for that lemma to apply.

4.6 The Impact of Edge Addition

In this section, we propose some techniques to help model the impact of edge addition on $t(G)$. As demonstrated in Corollary 3, we have a way of finding the seed graph, if one exists, for any k. If there exists a technique to find a seed for $k+1$ based on a seed for k, it would seem natural to contemplate the addition of an edge.

We generalize the notion of a spanning tree to include some other structures as follows: Given a graph G, a ***k-dispasugra***[3] is a spanning subgraph of G whose cyclespace has dimension k. Thus, a spanning tree is a *0-dispasugra*. Any spanning cycle is a *1-dispasugra*, but the converse is not true. Consider a *1-dispasugra* with cycle c. It is clear that by removing any edge from c we can obtain a *0-dispasugra*. In fact, this observation generalizes - for any *k-dispasugra*, removing any edge from any cycle results in a $(k-1)$-*dispasugra* due to the fact that the cyclespace dimension of a connected graph is $m-n+1$ [10] and removing any edge from any cycle in a *k-dispasugra* will result in a subgraph that is still connected and spanning.

Let $t_k(G)$ represent the number of *k-dispasugras* of a graph G. By definition, $t_0(G) = t(G)$, the number of spanning trees of G. Let $S_k(G)$ represent the set of *k-dispasugras* of a graph G. Let $C(G)$ be the set of all cycles of G and let $E(C(G))$ be the union of the edge sets of the cycles of $C(G)$. The number of cycles that an edge is in is $C(e,G)$. As a reminder from Section 1.2, to contract a cycle c in a graph G is to contract all the edges in the cycle c, which we denote with $G\cdot c$. Thus, the number of *k-dispasugra* that result from the contraction of cycle c is $t_k(G\cdot c)$.

Consider an (n,m) graph G with edge set $E = E(G)$ and let T be a spanning tree of G. For each edge $e \in E - E(T)$, the graph $T+e$ is a *1-dispasugra*. Thus, for each spanning tree of G there are $m-(n-1)$ ways to get a *1-dispasugra*. We can extend this notion. Thus, for *k-dispasugra* ρ, for each edge $e \in E - E(\rho)$, the graph $\rho + e$ is a

[3]The word *k-dispasugra* is a made up word that is a shortened form of *k-di*mension-*spa*nning-*su*b*gra*ph.

$(k+1)$-*dispasugra*. A k-*dispasugra* ρ has $(n-1)+k$ edges by definition. Thus, there are $m-(n-1)-k$ elements in $E-E(\rho)$, implying that there are $m-(n-1)-k$ ways we can add an edge to a k-*dispasugra* to get a $(k+1)$-*dispasugra*. This observation leads to the following equation:

$$(m-n+1-k)t_k(G) = \sum_{H \in S_{k+1}(G)} |E(C(H))| \tag{4.1}$$

The left-hand side of Equation 4.1 is the number of (k-*dispasugra*, additional edge) pairs in G. The right-hand side is the number of ($(k+1)$-*dispasugra*, edge from a cycle in the dispasugra) pairs in G. The right-hand side is the same as the left-hand side because any edge e from G not in a k-*dispasugra* ρ that is added to that ρ will be result in a $(k+1)$-*dispasugra* μ and that edge e will also be in a cycle in μ.

As a reminder, we use $C(e,G)$ to represent the set of cycles of G containing the edge e. Thus, we have $\sum_{e \in E(C(G))} |C(e,G)| = \sum_{c \in C(G)} |c|$ because each cycle contains $|c|$ edges. For each $e \in E(C(G))$, $c \in C(e,G)$ pair, find $t_k(G \cdot c)$ and add them up. Thus we get the following equation:

$$\sum_{c \in C(G)} |c| t_k(G \cdot c) = \sum_{e \in E(C(G))} \sum_{c \in C(e,G)} t_k(G \cdot c) \tag{4.2}$$

The left-hand side of Equation 4.2 is the number of (cycle, edge from cycle, k-*dispasugra*) triples which is the same as the number of (edge in a cycle, cycles containing that edge, k-*dispasugra*) triples, which is the right-hand side.

The following equation is a form of double-counting the number of (edge e, $(k+1)$-*dispasugra* involving edge e) tuples (on the left, the edge first and then the dispasugra, and on the right, for every such dispasugra, all the edges).

$$\sum_{e \in E(C(G))} t_{k+1}(G \cdot e) = \sum_{H \in S_{k+1}(G)} |E(C(H))| \tag{4.3}$$

The following equation is a consequence of the fact that every edge that is in a cycle in a $(k+1)$-*dispasugra* is in a cycle, and that the contraction of every such cycle results in some *k-dispasugra*.

$$\forall_{e \in E(C(G))} : t_{k+1}(G \cdot e) = \sum_{c \in C(e,G)} t_k(G \cdot c) \tag{4.4}$$

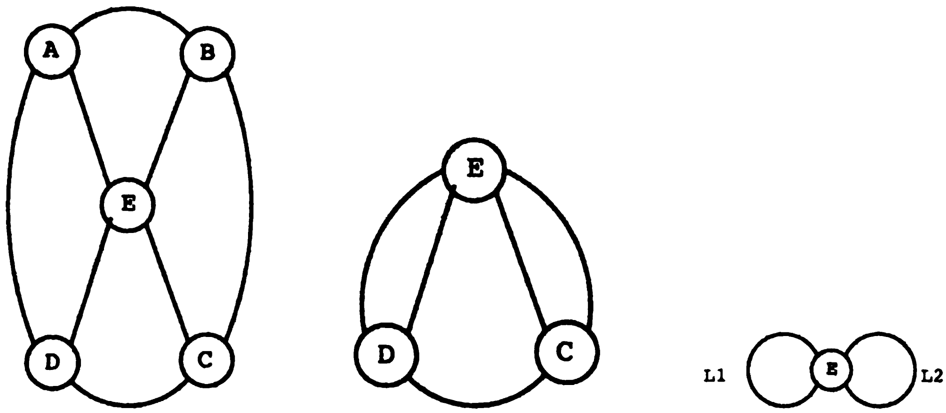
The following equation is merely the summation of the above equation over all the $e \in E(C(G))$.

$$\sum_{e \in E(C(G))} t_{k+1}(G \cdot e) = \sum_{e \in E(C(G))} \sum_{c \in C(e,G)} t_k(G \cdot c) \tag{4.5}$$

The following equation is the result of applying Equation 4.1, Equation 4.2, Equation 4.3, and Equation 4.5 together.

$$(m - n + 1 - k)t_k(G) = \sum_{c \in C(G)} |c| t_k(G \cdot c) \tag{4.6}$$

By a cycle reduction of a connected graph G we mean recursively contracting cycles (or loops or parallel edges), in some order, until we get a single vertex (or a tree if G is not biconnected). Note that the contraction of a cycle may result in loops or parallel edges, and that for our purposes, loops count as cycles as does a single pair of parallel edges. An ordered list of cycles which renders a cycle reduction of G will be referred to by β. Each cycle being contracted in a cycle reduction process is a cycle at the time it is contracted, but not necessarily prior to that. Note that not every list of cycles will result in a cycle reduction of G. When we contract a cycle, the label for the vertices that were part of the cycle are now all synonymous with the vertex that is the result of the contraction. We will use the notation $CR_\beta(G)$ to represent the cycle



(a) Graph G (b) $G_1 = G_0 \cdot \beta(1)$ (c) $G_2 = G_1 \cdot \beta(2)$

Figure 4.3: An example of two cycle contractions of a graph.

reduction of G with respect to a particular β where $CR_\beta(G) = \{G_0, G_1, \ldots, G_{|\beta|}\}$ such that G_0 is G and G_i is G_{i-1} with the ith element (cycle) of β having been contracted - that is, $G_i = G_{i-1} \cdot \beta_i$. Note that what is a cycle c in G_i may not be a cycle in G_j (where $j < i$) as it may involve two edges that are incident in G_i only because the vertex that they are both incident to is the result of the contraction of a previous cycle and that prior to such a contraction those edges were incident to different vertices.

Consider the graph G as found in Figure 4.3(a). An example β for G would be the ordered list $\{ABE, DEC, L1, L2\}$, where $L1$ and $L2$ are as in Figure 4.3(c). In $CR_\beta(G)$, $G_0 = G$ which is Figure 4.3(a). First, cycle ABE would be contracted, which yields graph G_1 in Figure 4.3(b). Then cycle DEC would be contracted, which yields graph G_2 in Figure 4.3(c). After that, cycle $L1$ (as found in Figure 4.3(c)) would be contracted resulting in a graph with a single vertex and loop for G_3, and finally cycle $L2$ (which also can be found in Figure 4.3(c)) would be contracted so that we result in a single vertex for G_4. The order in which the loops $L1$ and $L2$ are contracted matters as they are different cycles (as we are defining the notion of cycle) and a β is an ordering of these cycles, so order matters.

It can be shown that neither does an ordered basis of the cyclespace necessarily define a β nor does a β necessarily define an ordered basis of the cyclespace. Showing this is left as an exercise for the reader.

Let $\tau(G)$ be the set of all possible β for G. We can apply Equation 4.6 iteratively, contracting cycles along the way, until there are no more cycles (or loops) left in G. Since we contract all cycles, in all possible orders, we do this duplication of Equation 4.6 for all β in $\tau(G)$. For each β, we multiply the sizes of the cycles since, Equation 4.6, for each cycle contraction, results in the multiplication of that cycle's size. The factor on the left-hand side of Equation 4.6 is divided by on both sides and so brought over to the right-hand side to form the $|\beta|!$ found in the denominator in the resulting closed form which follows:

$$t(G) = t_0(G) = \sum_{\beta \in \tau(G)} \frac{\Pi_{c \in \beta} |c|}{|\beta|!} \tag{4.7}$$

Consider Equation 4.7. If we rewrite the cycle sizes as sums of chain lengths and then expand, we can see that this expression is the same as the formula in terms of chain lengths. The numerator will have every term in the chain length formula in every possible ordering and the denominator will have the number of possible orderings which will cancel out the fact that all possible orderings are represented.

Define a set of partial chains of a graph to be an edge disjoint set of paths such that every chain of the graph is expressible as a combination of some number of partial chains.

Lemma 14 *Equation 4.7 can be rewritten in terms of sums of partial chains.* □

Lemma 14 is true by definition of partial chains. To do so, just replace every chain with the sum of its partial chains.

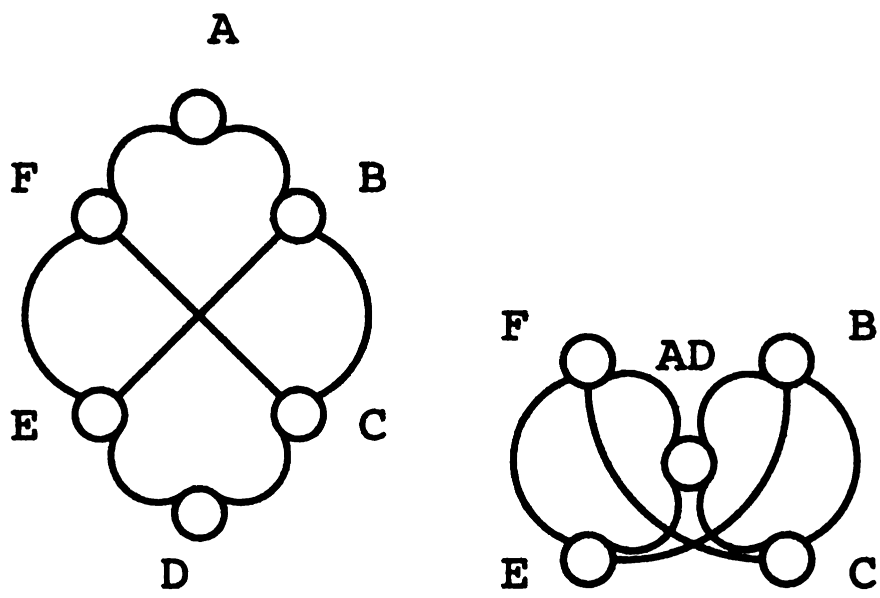
4.6.1 The Impact of Cojoining Vertices

We can cojoin two vertices of degree two that are end vertices of partial chains. This has an interpretation with respect to Lemma 14 as follows:

For each term in an expression in Lemma 14, the two vertices u and v to be cojoined will always be in the same cycle after the contraction of all the cycles prior to the last cycle y that contains both vertices to be cojoined as distinct vertices that have not already been contracted together. This is because each contraction of a cycle c involving a vertex v puts that vertex v into any cycle that had involved any of the other vertices on that cycle c. That last cycle will thus then be forced, due to the cojoining, to no longer be a cycle as both v and the vertex u that it is being cojoined with will be in the cycle, but cycles cannot have repeated vertices, and u and v are the same vertex after cojoining. If u and v are not both on y after the contraction of all cycles prior to y then, after the contraction of y, there is still a cycle which contains u and v as distinct vertices which contradicts the definition of y as the *last* cycle that contains u and v as distinct vertices.

Since u and v are in the same cycle after contraction of the prior cycles, the cycle can be split into two smaller cycles. As order matters (by the definition of β), there will be two new terms (one for each arrangement of the two smaller cycles), each of which will have a greater degree (a greater number of cycles). The two vertices to be cojoined will not be contracted into being the same vertex after the contraction of the prior cycles - if they were, then that means that they're both in a prior cycle c_p (after the contraction of the cycles prior to cycle c_p) and thus that is the factor in the term that should be affected.

So that we may keep track of which partial chains go into which cycle, when we express a cycle as a sum of partial chains, we always arrange the terms in order such that any two adjacent partial chain terms are next to each other in the algebraic expression. To facilitate finding out which terms involve which vertices, we subscript



(a) K_4 with two subdivisions

(b) A and D cojoined in Figure 4.4(a).

Figure 4.4: An example of the cojoining of two vertices.

every partial chain term with labels for the end vertices of that partial chain.

For example, the graph in Figure 4.4(a) has the terms found in Figure 4.5 in its expression for the number of spanning trees, using the previously described labeling notation, with the terms in lexicographic order. You will notice that only the first two terms have vertices A and D in the same cycle.

Now, if we cojoin vertices A and D as found in Figure 4.4(b) then we have Figure 4.6 as per the previous description of how to adjust the expression. Note that the expressions found in Figure 4.5 and Figure 4.6 are both expressions in terms of partial chain lengths. Thus, the determination of the number of spanning trees for a given distillation is the evaluation of the expression and is largely irrespective of the number of vertices or edges (as long as it is a graph that has that distillation).

For enough partial chains of sufficient length, the computation of the number of spanning trees through the Kirchoff matrix technique will be substantially more complex. The computational complexity for finding the determinant of a matrix is

$$
\begin{aligned}
&\tfrac{1}{3!}(x_{A,B}+x_{B,C}+x_{C,D}+x_{D,E}+x_{E,F}+x_{F,A})(x_{B,E})(x_{C,F})+\\
&\tfrac{1}{3!}(x_{A,B}+x_{B,C}+x_{C,D}+x_{D,E}+x_{E,F}+x_{F,A})(x_{C,F})(x_{B,E})+\\
&\tfrac{1}{3!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{C,D}+x_{D,E}+x_{B,E})(x_{E,F})+\\
&\tfrac{1}{3!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{B,E}+x_{E,F})(x_{C,D}+x_{D,E})+\\
&\tfrac{1}{3!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{C,D}+x_{D,E}+x_{E,F})(x_{B,E})+\\
&\tfrac{1}{3!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{B,C}+x_{C,F})(x_{C,D}+x_{D,E})+\\
&\tfrac{1}{3!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{C,D}+x_{D,E}+x_{C,F})(x_{B,C})+\\
&\tfrac{1}{3!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{B,C}+x_{C,D}+x_{D,E})(x_{C,F})+\\
&\tfrac{1}{3!}(x_{B,C}+x_{C,F}+x_{E,F}+x_{B,E})(x_{F,A}+x_{A,B})(x_{D,E}+x_{C,D})+\\
&\tfrac{1}{3!}(x_{B,C}+x_{C,F}+x_{E,F}+x_{B,E})(x_{D,E}+x_{C,D})(x_{F,A}+x_{A,B})+\\
&\tfrac{1}{3!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{C,F}+x_{E,F})(x_{A,B}+x_{F,A})+\\
&\tfrac{1}{3!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{A,B}+x_{F,A}+x_{C,F})(x_{E,F})+\\
&\tfrac{1}{3!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{A,B}+x_{F,A}+x_{E,F})(x_{C,F})+\\
&\tfrac{1}{3!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{B,E}+x_{B,C})(x_{A,B}+x_{F,A})+\\
&\tfrac{1}{3!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{B,E}+x_{A,B}+x_{F,A})(x_{B,C})+\\
&\tfrac{1}{3!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{B,C}+x_{A,B}+x_{F,A})(x_{B,E})
\end{aligned}
$$

Figure 4.5: An expression for the number of trees of Figure 4.4(a) where $x_{i,j}$ is the length of the partial chain from vertex i to vertex j.

at least order n^2 (and is typically closer to n^3). The number of arithmetic operations needed for the evaluation of the expression in Figure 4.6 is less than 300. Thus, if the partial chains were to be all of length three or greater, then clearly evaluation of the expressions found in Figure 4.5 and Figure 4.6 would be more efficient than utilizing the Kirchoff matrix technique. Utilization of the Kirchoff matrix technique would result in a matrix of larger than 20 by 20 and thus definitely take more computations than the technique described here. See such a matrix in Figure 4.7.

Consider the chain form of the Feussman formula:

$$t(G) = t(G-c)|c| + t(G\cdot c)$$

where c is a chain. This chain form of the formula is easily verified from the original Feussman formula ($t(G) = t(G-e) + t(G\cdot e)$) and it is easy to see that it is applicable to partial chains as well.

If we have a graph H and an edge $e = uv$ which is not in $E(H)$ and $G = H + e$ then $G\cdot e$ is H with the vertices u and v cojoined. We will denote the cojoining of

$$
\begin{aligned}
&\tfrac{1}{4!}(x_{A,B}+x_{B,C}+x_{C,D})(x_{D,E}+x_{E,F}+x_{F,A})(x_{B,E})(x_{C,F})+\\
&\tfrac{1}{4!}(x_{D,E}+x_{E,F}+x_{F,A})(x_{A,B}+x_{B,C}+x_{C,D})(x_{B,E})(x_{C,F})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,C}+x_{C,D})(x_{D,E}+x_{E,F}+x_{F,A})(x_{C,F})(x_{B,E})+\\
&\tfrac{1}{4!}(x_{D,E}+x_{E,F}+x_{F,A})(x_{A,B}+x_{B,C}+x_{C,D})(x_{C,F})(x_{B,E})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{C,D}+x_{D,E}+x_{B,E})(x_{E,F})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{B,E}+x_{E,F})(x_{C,D})(x_{D,E})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{B,E}+x_{E,F})(x_{D,E})(x_{C,D})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{C,D})(x_{D,E}+x_{E,F})(x_{B,E})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,C}+x_{C,F}+x_{F,A})(x_{D,E}+x_{E,F})(x_{C,D})(x_{B,E})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{B,C}+x_{C,F})(x_{C,D})(x_{D,E})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{B,C}+x_{C,F})(x_{D,E})(x_{C,D})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{C,D})(x_{D,E}+x_{C,F})(x_{B,C})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{D,E}+x_{C,F})(x_{C,D})(x_{B,C})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{B,C}+x_{C,D})(x_{D,E})(x_{C,F})+\\
&\tfrac{1}{4!}(x_{A,B}+x_{B,E}+x_{E,F}+x_{F,A})(x_{D,E})(x_{B,C}+x_{C,D})(x_{C,F})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,F}+x_{E,F}+x_{B,E})(x_{F,A}+x_{A,B})(x_{D,E})(x_{C,D})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,F}+x_{E,F}+x_{B,E})(x_{F,A}+x_{A,B})(x_{C,D})(x_{D,E})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,F}+x_{E,F}+x_{B,E})(x_{D,E}+x_{C,D})(x_{F,A})(x_{A,B})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,F}+x_{E,F}+x_{B,E})(x_{D,E}+x_{C,D})(x_{A,B})(x_{F,A})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{C,F}+x_{E,F})(x_{A,B})(x_{F,A})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{C,F}+x_{E,F})(x_{F,A})(x_{A,B})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{A,B})(x_{F,A}+x_{C,F})(x_{E,F})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{F,A}+x_{C,F})(x_{A,B})(x_{E,F})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{A,B})(x_{F,A}+x_{E,F})(x_{C,F})+\\
&\tfrac{1}{4!}(x_{B,C}+x_{C,D}+x_{D,E}+x_{B,E})(x_{F,A}+x_{E,F})(x_{A,B})(x_{C,F})+\\
&\tfrac{1}{4!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{B,E}+x_{B,C})(x_{A,B})(x_{F,A})+\\
&\tfrac{1}{4!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{B,E}+x_{B,C})(x_{F,A})(x_{A,B})+\\
&\tfrac{1}{4!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{B,E}+x_{A,B})(x_{F,A})(x_{B,C})+\\
&\tfrac{1}{4!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{F,A})(x_{B,E}+x_{A,B})(x_{B,C})+\\
&\tfrac{1}{4!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{B,C}+x_{A,B})(x_{F,A})(x_{B,E})+\\
&\tfrac{1}{4!}(x_{C,D}+x_{D,E}+x_{E,F}+x_{C,F})(x_{F,A})(x_{B,C}+x_{A,B})(x_{B,E})
\end{aligned}
$$

Figure 4.6: An expression for the number of trees of Figure 4.4(b) where $x_{i,j}$ is the length of the partial chain from vertex i to vertex j.

$$
\begin{bmatrix}
2 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & 3 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\
0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\
0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & -1 \\
-1 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2
\end{bmatrix}
$$

Figure 4.7: A matrix

vertices u and v of H via $H \cdot uv$. Cojoining can be thought of as the contraction of a non-existent edge.

Thus, we can think of adding a chain c to a graph G as the following:

$$t(G) = t(G - c)|c| + t(G \cdot c)$$

We can apply the above regardless of whether or not c is a chain in G or not - if not, the internal vertices of the chain (if any) must be new vertices and the end vertices must be vertices that already exist in G. The $G \cdot c$ represents the cojoining. The $t(G - c)|c|$ represents the unaffected terms, which means the chain c is a cycle after the contraction of all other edges.

Thus, we can use the cycle contraction formula and the Feussman formula to find a new chain form of the formula. There is a way we can use the cycle contraction formula and the Feussman formula to obtain a cycle contraction formula (without needing to factor the chain form of the formula) and we describe it next.

4.6.2 The Reverse Application of The Feussman Formula to A Cycle Contraction Formula

Every chain and every edge is represented in every β, since every β results in the contraction of every cycle - if a chain or edge were not contracted then β would not reduce the graph down to a single vertex.

Consider the application of Equation 4.7 to Equation 4.6.

$$(m-n+1-k)t(G) = \sum_{c \in C(G)} |c| \sum_{\beta \in \tau(G \cdot c)} \frac{\Pi_{c \in \beta}|c|}{|\beta|!} \tag{4.8}$$

This shows that we can factor out any cycle. For every β, there is a cycle $\mathscr{C}$ that contains partial chain ϕ. Let us factor out that cycle $\mathscr{C}$. For every such ϕ containing $\mathscr{C}$ factored β, we may apply the Feussman formula with regard to ϕ. This use of the Feussman formula in the context of a cycle contraction formula is something that, similar to the addition of a chain to a chain length formula to find a new chain length formula, can be applied in reverse, with the caveat that every new β must be a valid β and that no β be overlooked.

To ensure that every new β is a valid β it suffices to ensure that ϕ only be added to terms for which it is part of that corresponding cycle, which may include previously empty cycles (previously empty terms). By "previously empty cycle", we mean a cycle c containing the end vertices of this new chain, (but not the edges of the new chain) all of whose (c's) edges have been contracted in prior cycles in the β.

To ensure that no β is overlooked, it suffices to show that consideration of the addition of ϕ to all terms (including the empty ones implicit between all other terms) will cover all possible β. This is clearly the case as every ϕ is either in a term by itself (an empty term) or not (an already existant term).

Thus, for a given cycle contraction formula for a graph G, we can add a chain between any two vertices that are partial chain endpoints and get a new cycle contraction formula. Thus, we have a way to model the impact of chain addition on the expression for the number of spanning trees in terms of chain lengths when in the cycle contraction formula form.

Chapter 5

Future Work

In the future, we would like to prove or at least further examine many of the conjectures mentioned. Amongst those conjectures would be those found in Section 3.4 and this section. We would also like to further investigate the lower bounds possible through the techniques described in Section 3.5.

Previous algorithms for finding t-optimal (n, m) graphs when $m \leq n + 3$ have been published, as have formulas for $t(n, m)$ when $m \leq n+2$. We have an algorithm that is accurate within certain bounds, when $m \leq \frac{3n}{2}$ and $k \geq 2$, for $(n, n+k)$, which provides us with polynomial formulas in some cases and bounds on leading coefficients of these polynomials on all others. This not only gives us an explicit formula for the $(n, n+3)$ case and the distillation of the $(n, n+4)$ case, but, up to computational feasibility of algorithm execution, all $m \leq \frac{3n}{2}$ cases.

We also have another conjecture, which all observed $(n, n+k)$ follow, namely, Conjecture 6.

Conjecture 6 *A technique to create 3-regular graphs with large numbers of spanning trees:*

- *Find an optimal $(n, n+k)$ graph for some n (where $n \geq 2k$).*
- *Find the distillation and then iterate Algorithm 1 twice.*

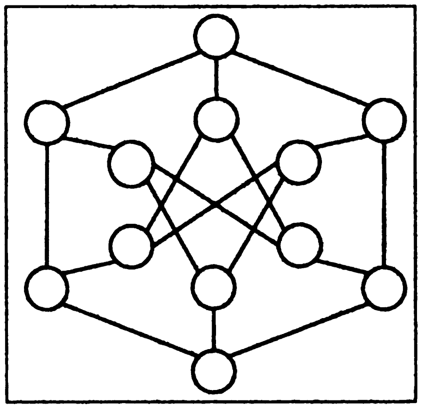


Figure 5.1: The t-optimal $(12, 18)$ graph

- *There will be two non-adjacent vertices of degree 2; join them with a new edge.*
- *You now have the distillation of a t-optimal $(n, n+k+1)$ graph.*

Conjecture 6 holds for those cases we have seen so far, such as the simple cycle, θ graph, K_4, $K_{3,3}$, and twisted 3-cube. Based on this conjecture, the optimal distillations for $(n, n+5)$ and $(n, n+6)$ are the following[1], respectively: the Petersen graph, and then the $(12, 18)$ graph found in Figure 5.1.

In addition to Conjecture 6 and the other conjectures that we've mentioned earlier, we also conjecture that all t-optimal (n, m) graphs with $m > \frac{3n}{2}$ can be found by taking a three-regular t-optimal graph and contracting a certain spanning forest in which the tree sizes differ by no more than one.

As we now have a means to better, more efficiently, and more concisely understand the impact of edge operations such as addition, deletion, subdivision, and contraction on the number of spanning trees of a graph, we now plan on more thoroughly investigating some of these conjectures, specifically Conjecture 6.

[1] The distillations are t-optimal graphs, we have verified this through the use of brute force already. It remains to be seen if they are the distillations of all t-optimal $(n, n+k)$ graphs with the same k where $k \geq \frac{n}{2}$.

Appendix A

Figures

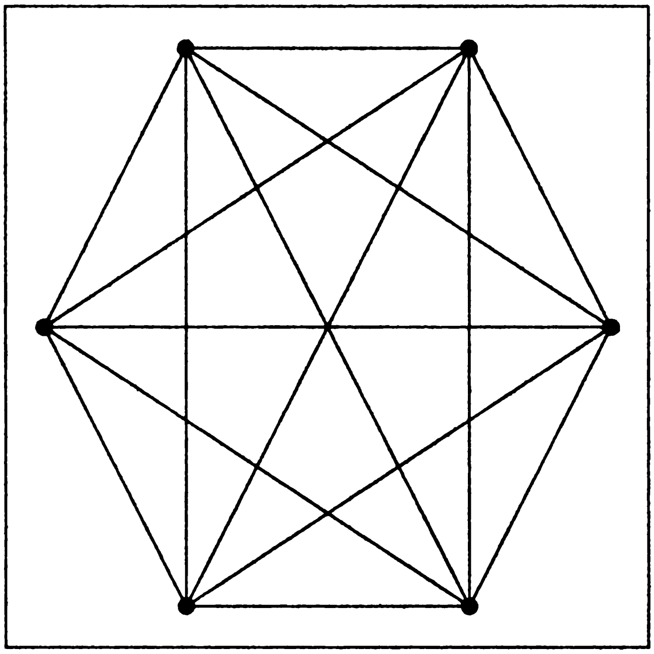


Figure A.1: A complete 5-regular 1-partite graph.

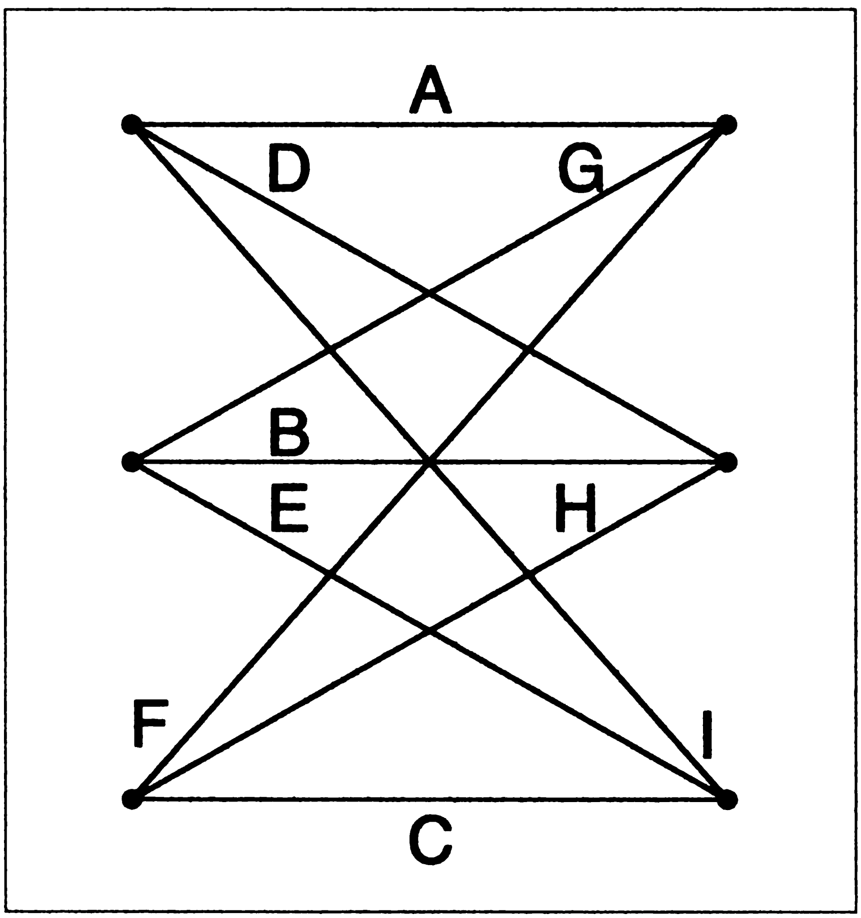


Figure A.2: A complete 3-regular 2-partite graph.

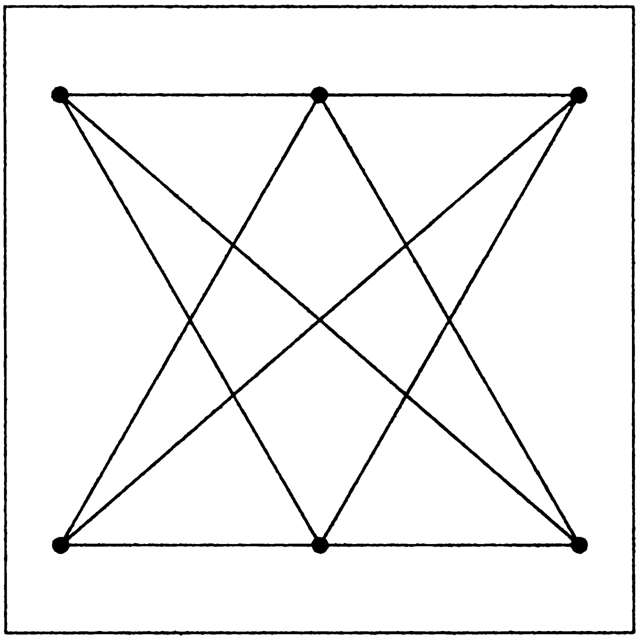


Figure A.3: A complete 4-regular 3-partite graph. Note how every vertex is connected to any vertex not in the same partition.

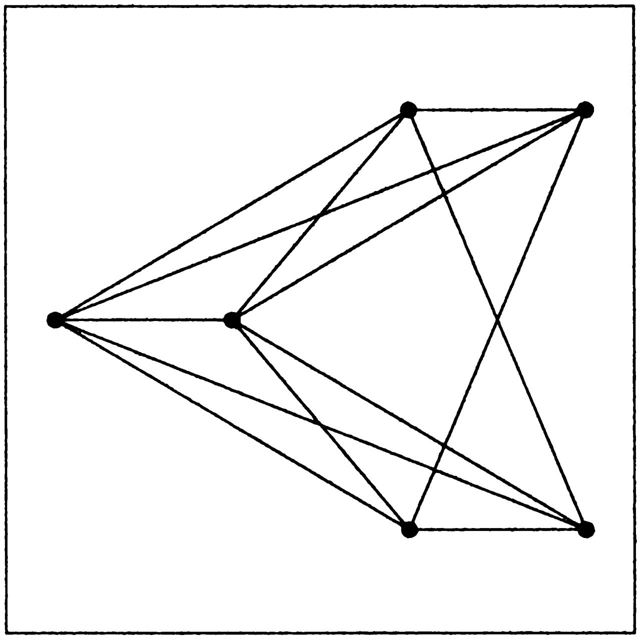


Figure A.4: A complete $4-5$ regular 4-partite graph.
This graph is $K_{1,1,2,2}$.

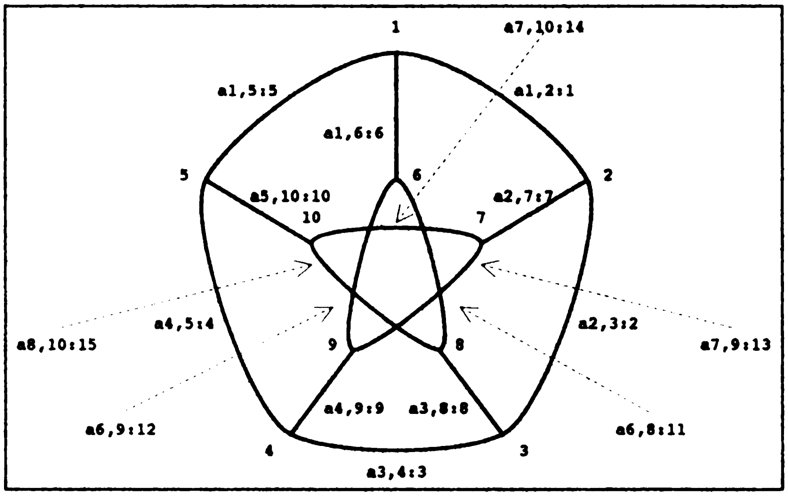


Figure A.5: The Petersen graph, a t-optimal graph with 10 vertices and 15 edges.

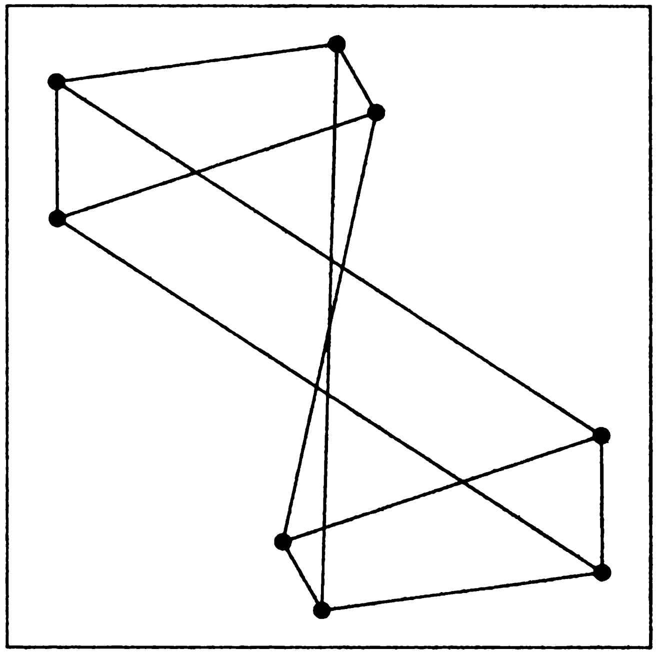


Figure A.6: A spring embedding of the t-optimal graph with 8 vertices and 12 edges. This graph is more commonly drawn as in Figure 3.1.

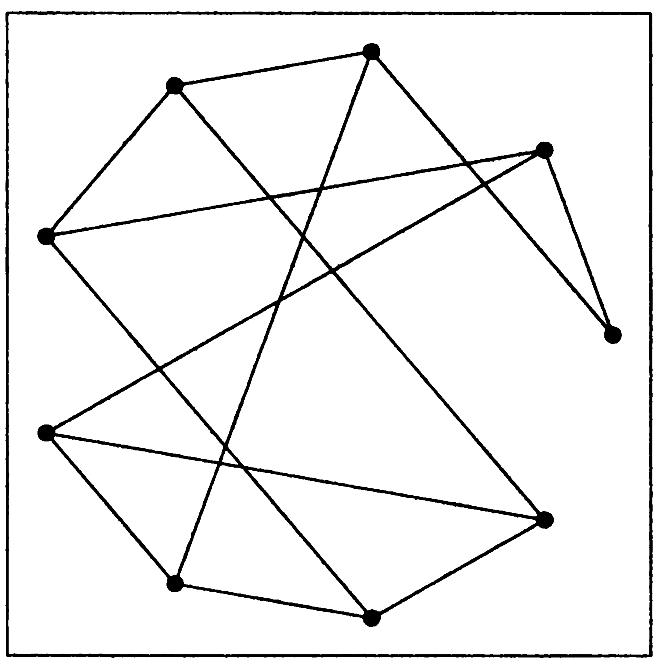


Figure A.7: An embedding of the t-optimal graph with 9 vertices and 13 edges

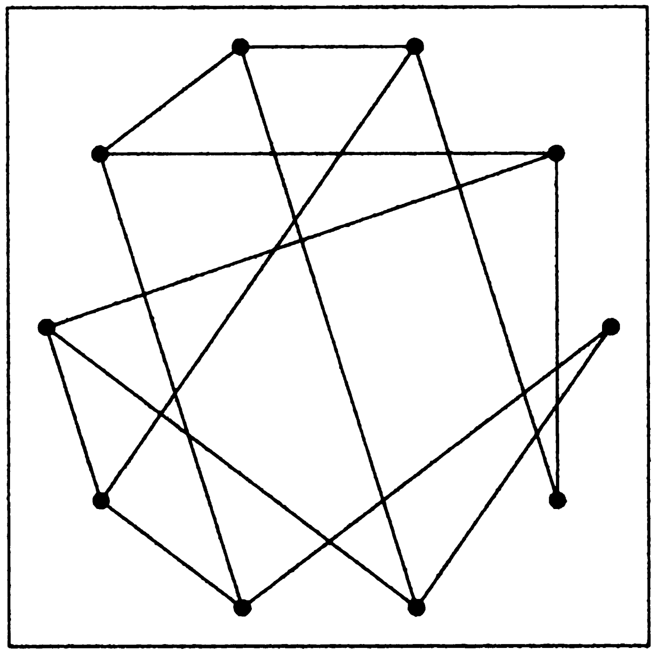


Figure A.8: An embedding of the t-optimal graph with 10 vertices and 14 edges

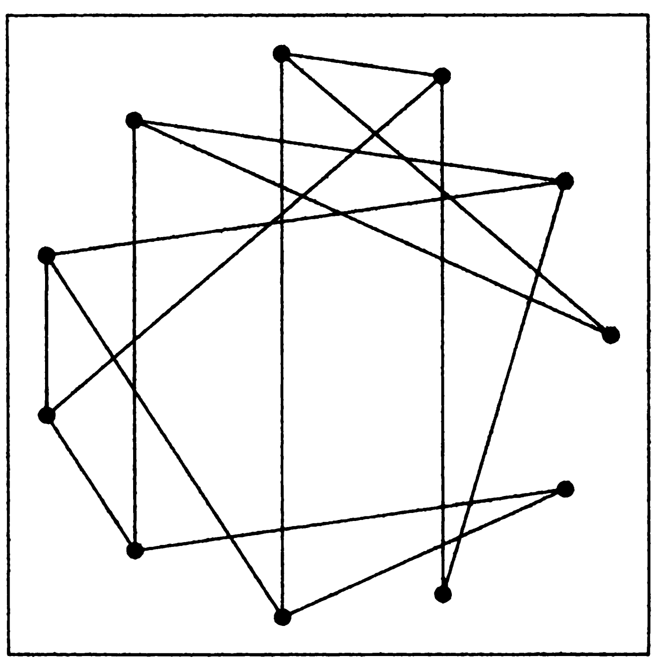


Figure A.9: An embedding of the t-optimal graph with 11 vertices and 15 edges

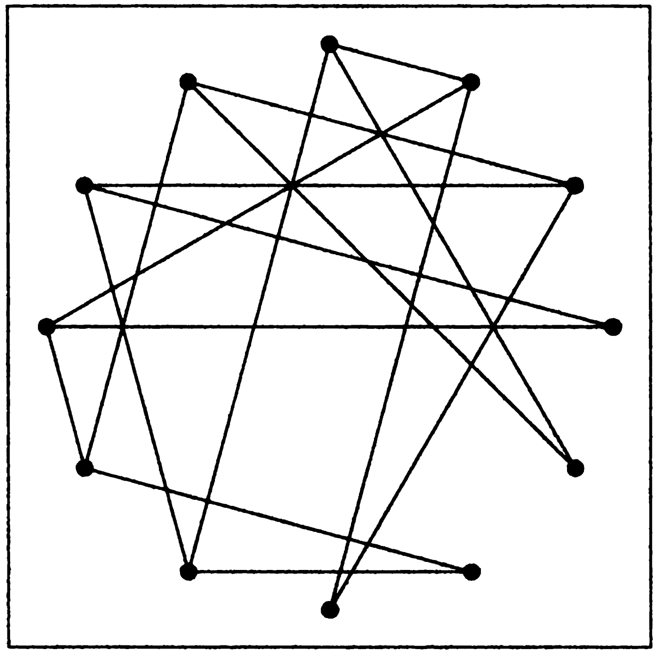


Figure A.10: An embedding of the t-optimal graph with 12 vertices and 16 edges

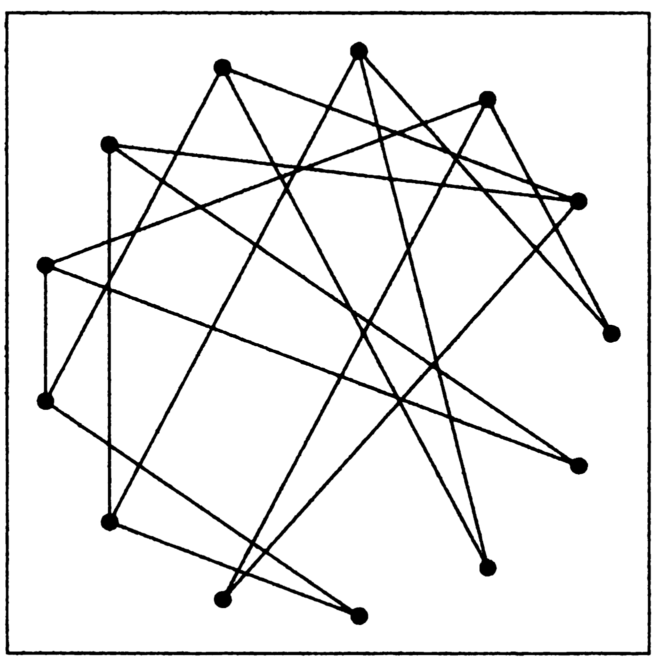


Figure A.11: An embedding of the t-optimal graph with 13 vertices and 17 edges

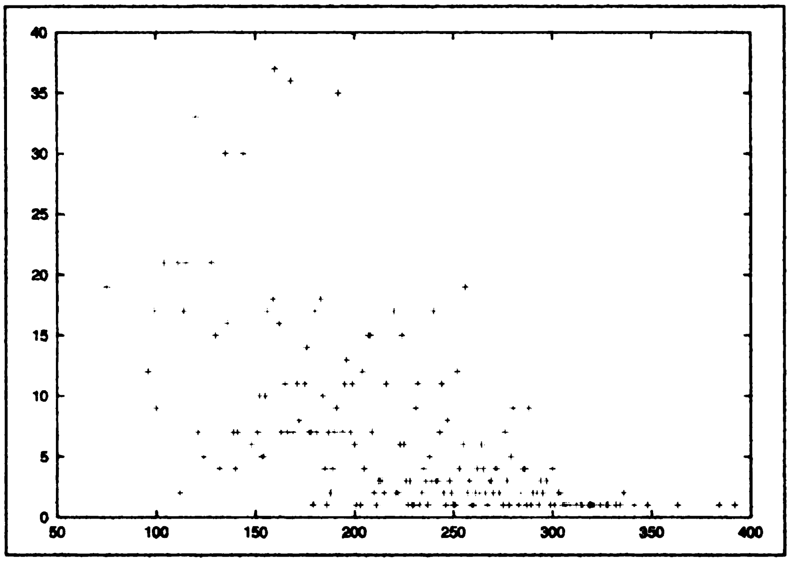


Figure A.12: A histogram of non-isomorphic graphs and their number of spanning trees with 8 vertices and 12 edges. The x axis is the number of spanning trees. The y axis is the number of non-isomorphic graphs with that number of spanning trees

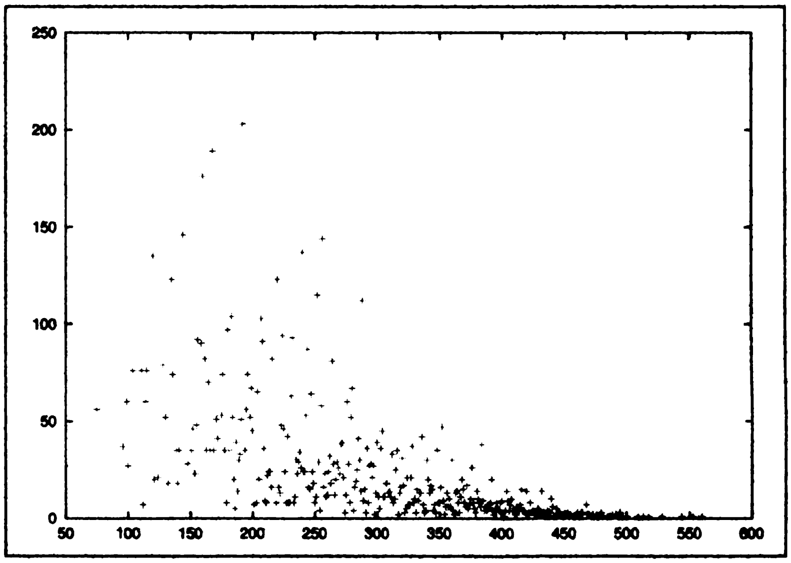


Figure A.13: A histogram of non-isomorphic graphs and their number of spanning trees with 9 vertices and 13 edges. The x axis is the number of spanning trees. The y axis is the number of non-isomorphic graphs with that number of spanning trees

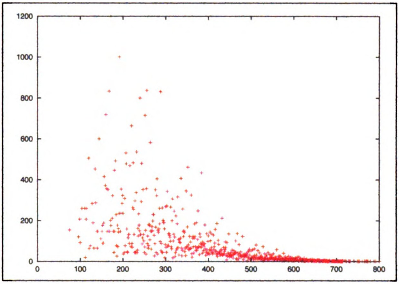


Figure A.14: A histogram of non-isomorphic graphs and their number of spanning trees with 10 vertices and 14 edges. The x axis is the number of spanning trees. The y axis is the number of non-isomorphic graphs with that number of spanning trees

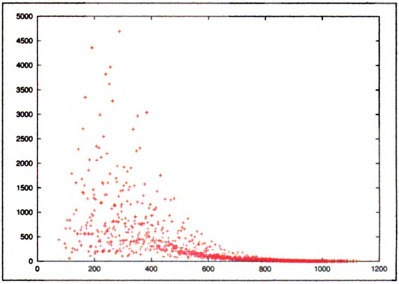
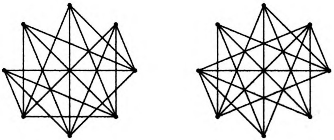


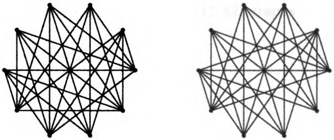
Figure A.15: A histogram of non-isomorphic graphs and their number of spanning trees with 11 vertices and 15 edges. The x axis is the number of spanning trees. The y axis is the number of non-isomorphic graphs with that number of spanning trees



(a) A t-optimal $(8, 18)$ graph

(b) Another t-optimal $(8, 18)$ graph

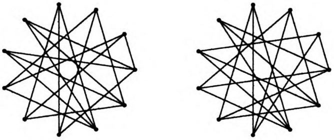
Figure A.16: t-optimal $(8, 18)$ graphs



(a) A t-optimal $(10, 27)$ graph

(b) Another t-optimal $(10, 27)$ graph

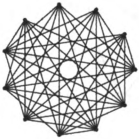
Figure A.17: t-optimal $(10, 27)$ graphs



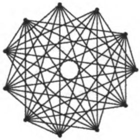
(a) A t-optimal $(11, 18)$ graph

(b) Another t-optimal $(11, 18)$ graph

Figure A.18: t-optimal $(11, 18)$ graphs

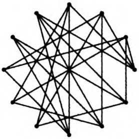


(a) A t-optimal $(11, 42)$ graph

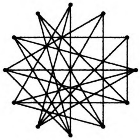


(b) Another t-optimal $(11, 42)$ graph

Figure A.19: t-optimal $(11, 42)$ graphs



(a) A t-optimal $(12, 22)$ graph



(b) Another t-optimal $(12, 22)$ graph

Figure A.20: t-optimal $(12, 22)$ graphs

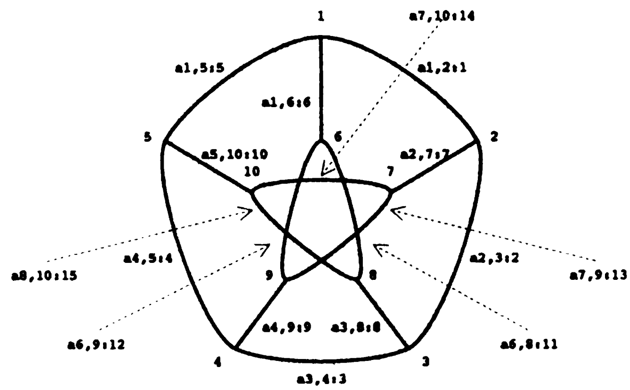


Figure A.21: The result of applying Algorithm 1 0 times to the Petersen graph.

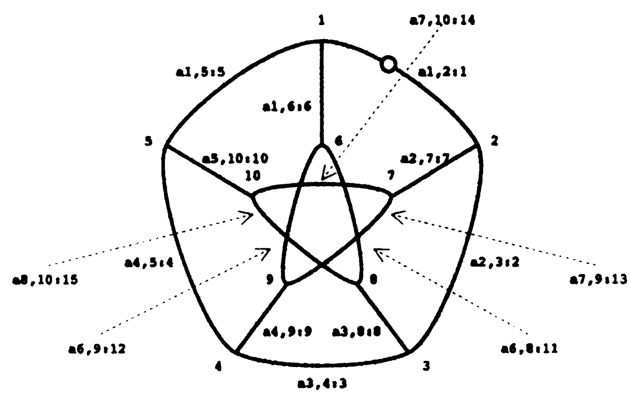


Figure A.22: The result of applying Algorithm 1 1 times to the Petersen graph.

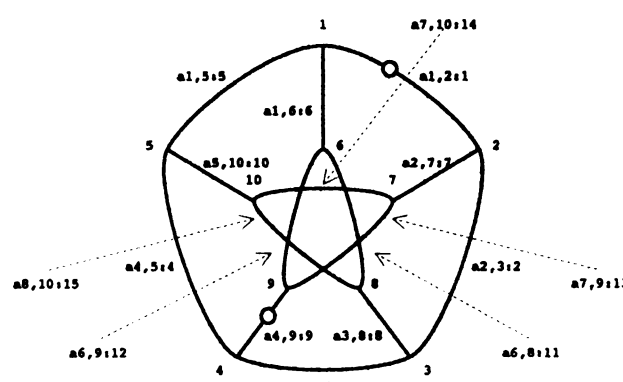


Figure A.23: The result of applying Algorithm 1 2 times to the Petersen graph.

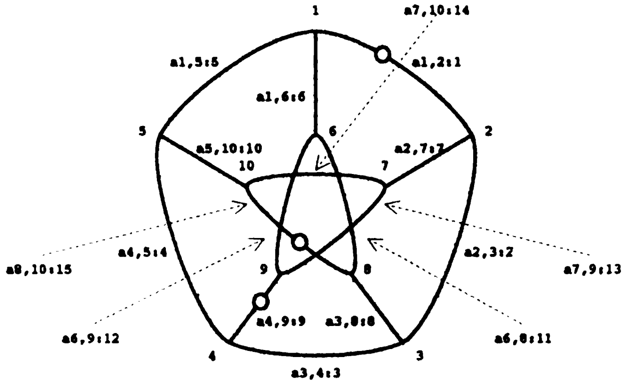


Figure A.24: The result of applying Algorithm 1 3 times to the Petersen graph.

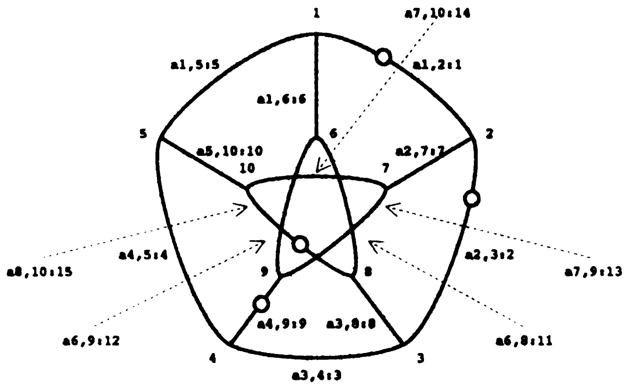


Figure A.25: The result of applying Algorithm 1 4 times to the Petersen graph.

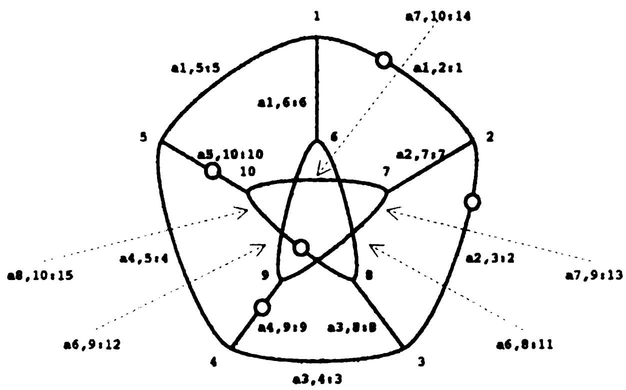


Figure A.26: The result of applying Algorithm 1 5 times to the Petersen graph.

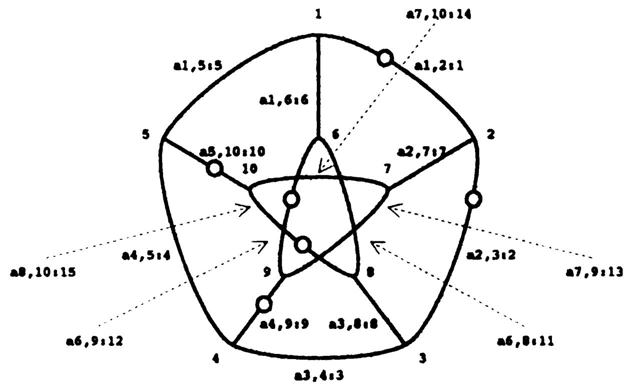


Figure A.27: The result of applying Algorithm 1 6 times to the Petersen graph.

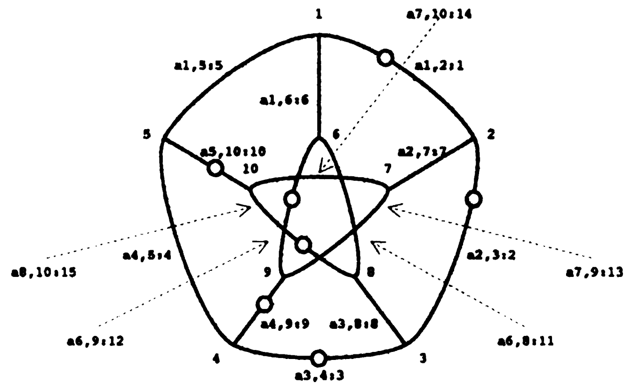


Figure A.28: The result of applying Algorithm 1 7 times to the Petersen graph.

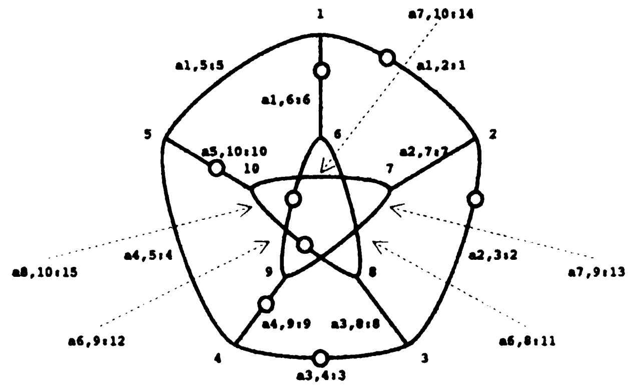


Figure A.29: The result of applying Algorithm 1 8 times to the Petersen graph.

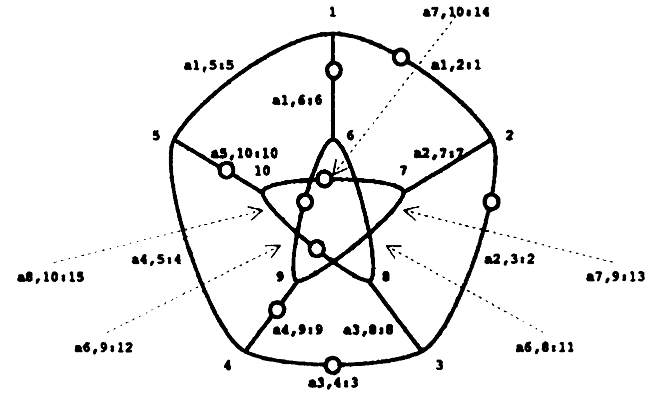


Figure A.30: The result of applying Algorithm 1 9 times to the Petersen graph.

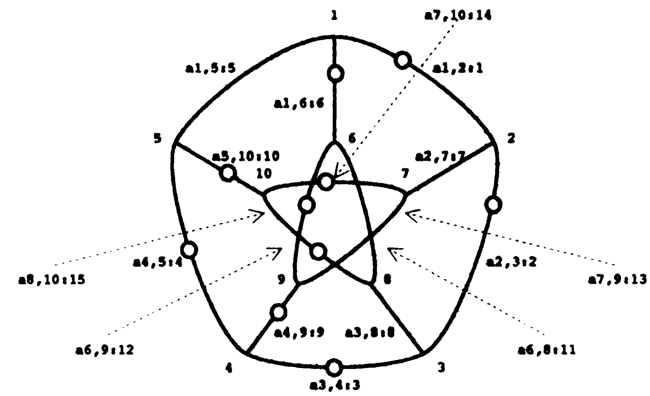


Figure A.31: The result of applying Algorithm 1 10 times to the Petersen graph.

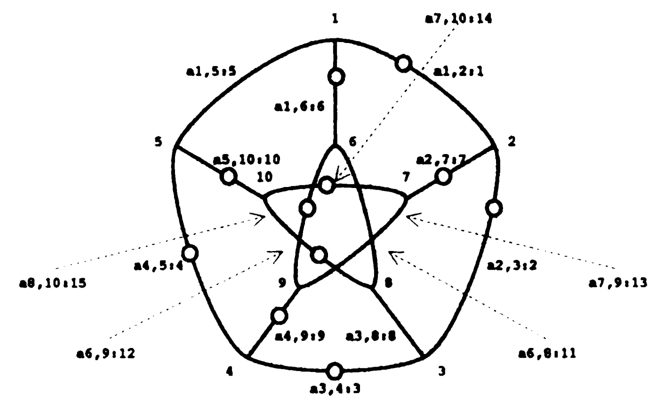


Figure A.32: The result of applying Algorithm 1 11 times to the Petersen graph.

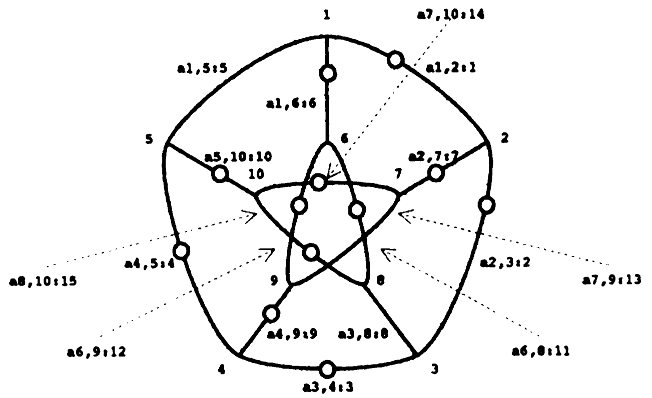


Figure A.33: The result of applying Algorithm 1 12 times to the Petersen graph.

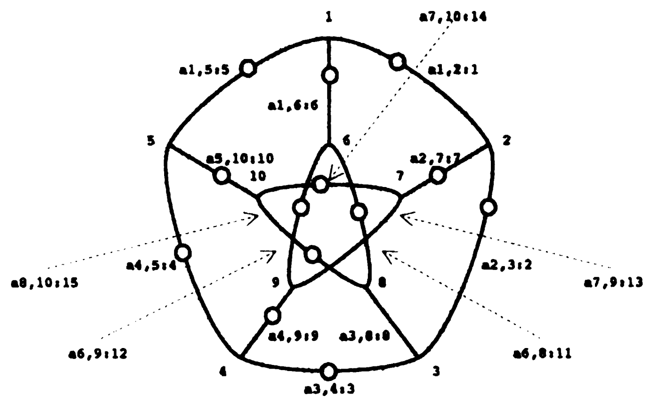


Figure A.34: The result of applying Algorithm 1 13 times to the Petersen graph.

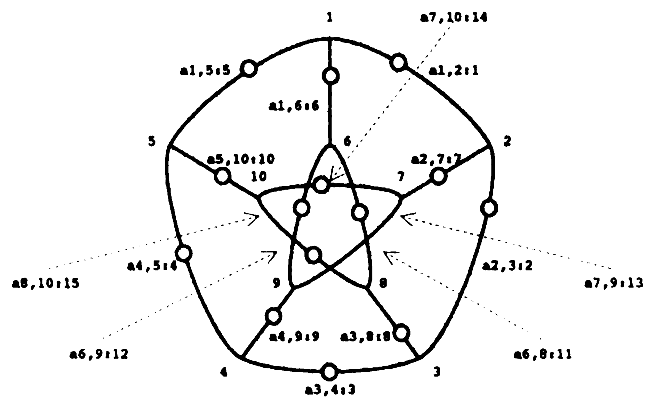


Figure A.35: The result of applying Algorithm 1 14 times to the Petersen graph.

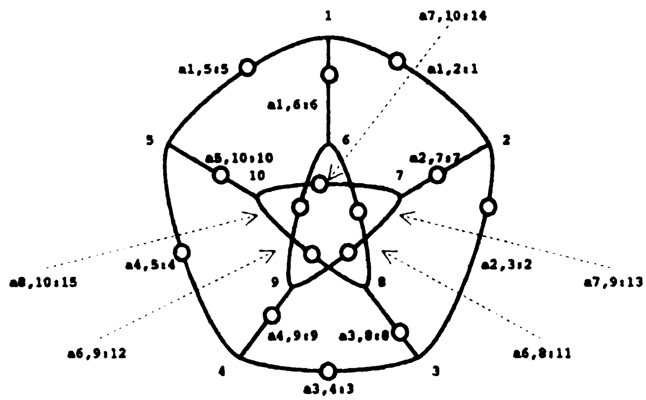


Figure A.36: The result of applying Algorithm 1 15 times to the Petersen graph.

Appendix B

Tables

m	$n=3$	4	5	6	7	8	9	10	11	12
3	3									
4		4								
5		8	5							
6		16	12	6						
7			24	16	7					
8			45	36	21	8				
9			75	81	51	27	9			
10			125	135	117	72	33	10		
11				225	231	168	96	40	11	
12				384	432	392	240	128	48	12
13				576	720	720	560	328	160	56
14				864	1200	1280	1200	800	448	200
15				1296	1840	2304	2223	2000	1120	600
16					2800	4096	4032	3672	2800	1568
17					4200	6144	7168	7000	6020	3920
18					6125	9216	12480	12584	11760	9800
19					8575	13825	19760	22720	21952	19208
20					12005	21025	32000	40960	40365	37632

Table B.1: A table of some of the results we obtained. (I)
The entries are the values of $t(n, m)$.

m	$n=7$	$n=8$	$n=9$	$n=10$	$n=11$	$n=12$
21	16807	30000	48000	64125	72657	68992
22		42000	72000	101250	130691	127764
23		58800	105000	159375	208320	233037
24		82944	148625	250000	332800	
25		110592	210125	390625	521284	
26		147456	295488	546875	810000	
27		196608	419904	765625	1260000	
28		262144	559872	1071875	1878000	
29			746496	1500625	2740500	
30			1000188	2116800	4050000	
31			1333584	2861568	5670000	
32			1750329	3852576	7938000	
33			2250423	5186160	11113200	
34			2893401	6914880	14929920	
35			3720087	9219840	20030976	
36			4782969	11908960	26873856	
37				15366400	35831808	
38				19756800	48185669	
39				25401600	63518455	
40				32768000	82824896	
41				40960000	106489152	
42				51200000	136914624	
43				64000000	176046080	
44				80000000	227127296	
45				100000000	285474816	
46					356843520	
47					446054400	
48					558892224	
49					698615280	
50					864536409	
51					1056655611	
52					1291467969	
53					1578460851	
54					1929229929	
55					1937019605	

Table B.2: A table of some of the results we obtained. (II)
The entries are the values of $t(n, m)$.

chain: 1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	$t(G)$	Figure
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2000	A.21
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2800	A.22
2	1	1	1	1	1	1	1	2	1	1	1	1	1	1	3920	A.23
2	1	1	1	1	1	1	1	2	1	1	1	1	1	2	5488	A.24
2	2	1	1	1	1	1	1	2	1	1	1	1	1	2	7448	A.25
2	2	1	1	1	1	1	1	2	2	1	1	1	1	2	10101	A.26
2	2	1	1	1	1	1	1	2	2	1	2	1	1	2	13689	A.27
2	2	2	1	1	1	1	1	2	2	1	2	1	1	2	18096	A.28
2	2	2	1	1	2	1	1	2	2	1	2	1	1	2	23868	A.29
2	2	2	1	1	2	1	1	2	2	1	2	1	2	2	31212	A.30
2	2	2	2	1	2	1	1	2	2	1	2	1	2	2	39984	A.31
2	2	2	2	1	2	2	1	2	2	1	2	1	2	2	51200	A.32
2	2	2	2	1	2	2	1	2	2	2	2	1	2	2	65536	A.33
2	2	2	2	2	2	2	1	2	2	2	2	1	2	2	81920	A.34
2	2	2	2	2	2	2	2	2	2	2	2	1	2	2	102400	A.35
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	128000	A.36

Table B.3: Tuples of chain lengths, number of spanning trees, and diagram figures for t-optimal graphs with distillation and labeling as found in Figure A.5.

Bibliography

[1] B. D. McKay, *nauty user's guide (version 1.5), Technical report TR-CS-90-02.* Australian National University, Computer Science Department, ANU, 1990.

[2] D. Cvetkovic, M. Doob, and H. Sachs, *Spectra of Graphs.* VEB Deutscher Verlag der Wissenchafen, Berlin, 1982.

[3] A. K. Kel'mans, "The number of trees in a graph i.," *Automation and Remote Control - Translated from: Avtomatika i Telemekhanika (Russian)*, vol. 26, no. 12, pp. 2194–2204, 1965.

[4] D. R. Shier, "Maximizing the number of spanning trees in a graph with n nodes and m edges," *J. Res. Nat. Bur. Stand., Sect. B*, vol. 78, pp. 193–196, 1974.

[5] A. K. Kel'mans and V. M. Chelnokov, "A certain polynomial of graphs and graphs with an extremal number of trees," *Journal of Combinatorial Theory, Series B*, vol. 16, pp. 197–214, 1974.

[6] L. Petingi, F. Boesch, and C. Suffel, "On the characterization of graphs with maximum number of spanning trees," *Discrete Mathematics*, 1997.

[7] L. Petingi, *On the characterization of graphs with maximum number of spanning trees.* PhD thesis, Stevens Institute of Technology, Hoboken, NJ, 1991.

[8] L. Petingi and J. Rodriguez, "A new technique for the characterization of graphs with a maximum number of spanning trees," *Special Issue on Algebraic and Topological Methods in Graph Theory. Discrete Mathematics*, vol. 244, pp. 351–373, 2002.

[9] F. T. Boesch, X. Li, and C. Suffel, "On the existence of uniformly optimally reliable networks," *Networks*, vol. 21, pp. 181–194, 1991.

[10] D. B. West, *Introduction to Graph Theory.* Prentice Hall, Upper Saddle River, 1996.

[11] W. Feussner, "Zur berechung der stromstarke in netzformigen leitern," *Ann. Phys.*, vol. 15, pp. 385–394, 1904.

[12] J. F. Wang and M. H. Wu, "Network reliability analysis: On maximizing the number of spanning trees," *Proceed. Nat. Sci. Coun. Repub. China, Part A. Phys. Sci. Eng.*, vol. 11, no. 3, pp. 193–196, 1987.

[13] S. S. Tseng and L. R. Wang, "Maximizing the number of spanning trees of networks based on cycle basis representation," *Int. J. Comput. Math*, vol. 28, pp. 49–56, 1989.

[14] G. Wang, "A Proof of Boesch's Conjecture," *Networks*, vol. 24, pp. 277–284, 1994.

[15] B. Gilbert and W. Myrvold, "Maximixing spanning trees in almost complete graphs," *Networks*, vol. 30, pp. 23–30, 1997.

[16] A. Cayley, "A theorem on trees," *Quarterly Journal of Mathematics*, vol. 23, pp. 376–378, 1889.

[17] G. Baron, H. Prodinger, R. F. Tichy, F. T. Boesch, and J. F. Wang, "The number of spanning trees in the square of a cycle," *Fibonnacci Journal*, pp. 258–264, August 1985.

[18] Z. Lonc, K. Parol, and J. M. Wojciechowski, "On the asymptotic behavior of the maximum number of spanning trees in circulant graphs," *Networks*, vol. 30, pp. 47–56, January 1997.

[19] C.-S. Cheng, "Optimality of certain asymmetrical experimental designs," *Annual of Statistics*, vol. 6, no. 6, pp. 1239–1261, 1978.

[20] C.-S. Cheng, "Maximizing the total number of spanning trees in a graph: Two related problems in graph theory and optimum design theory," *Journal of Combinatorial Theory, Series B*, vol. 31, pp. 240–248, 1981.

[21] L. Petingi and J. Rodriguez, "On the conjecture that among all graphs on n nodes and e edges, one with maximum number of spanning trees must be almost-regular," *Congressus Numerantium*, vol. 136, pp. 207–214, 1999.

[22] A.-H. Esfahanian, L. M. Ni, and B. Sagan, "The twisted n-cube with application to multiprocessing," *IEEE Transactions on Computers*, vol. 40, pp. 88–94, January 1991.

[23] A. Chen and A.-H. Esfahanian, "A demography of t-optimal (n, m) graphs where $n \leq 12$," *Proceedings of The 2005 International Conference on Algorithmic Mathematics and Computer Science, 2nd Edition*, 2005.

[24] A. Chen and A.-H. Esfahanian, "A demography of t-optimal (n, m) graphs where $n \leq 12$," Tech. Rep. MSU-CSE-04-50, Computer Science and Engineering, Michigan State University, East Lansing, Michigan, December 2004.