uOttawa

L'Université canadienne
Canada's university

FACULTÉ DES ÉTUDES SUPÉRIEURES ET POSTOCTORALES

uOttawa

L'Université canadienne
Canada's university

FACULTY OF GRADUATE AND POSDOCTORAL STUDIES

Paul Elliot-Magwood
AUTEUR DE LA THÈSE / AUTHOR OF THESIS

Ph.D. (Mathematics)
GRADE / DEGREE

Department of Mathematics and Statistics
FACULTÉ, ÉCOLE, DÉPARTEMENT / FACULTY, SCHOOL, DEPARTMENT

The Integrality Gap of the Asymmetric Traveling Salesman Problem

TITRE DE LA THÈSE / TITLE OF THESIS

Sylvia Boyd
DIRECTEUR (DIRECTRICE) DE LA THÈSE / THESIS SUPERVISOR

CO-DIRECTEUR (CO-DIRECTRICE) DE LA THÈSE / THESIS CO-SUPERVISOR

EXAMINATEURS (EXAMINATRICES) DE LA THÈSE / THESIS EXAMINERS

Kevin Cheung

Bruce Shepherd

Lucia Moura

Brett Stevens

Gary W. Slater
Le Doyen de la Faculté des études supérieures et postdoctorales / Dean of the Faculty of Graduate and Postdoctoral Studies

THE INTEGRALITY GAP OF THE ASYMMETRIC TRAVELLING SALESMAN PROBLEM

Paul Elliott-Magwood

Thesis Submitted to the Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
for the degree of
Doctor of Philosophy in Mathematics[1]

Department of Mathematics and Statistics
Faculty of Science
University of Ottawa

[1]The Ph.D. Program is a joint program with Carleton University, administered by the Ottawa-Carleton Institute of Mathematics and Statistics

Your file Votre référence
ISBN: 978-0-494-50727-8
Our file Notre référence
ISBN: 978-0-494-50727-8

Abstract

The Asymmetric Travelling Salesman Problem (ATSP) is a famous problem in the field of Combinatorial Optimization with many applications in industry. This problem is known to be NP-hard; moreover, no constant factor polytime approximation algorithm is known for solving the ATSP even when the costs are metric. In this thesis we study the integrality gap for the ATSP, i.e. the worst-case ratio between the ATSP and its linear programming relaxation ATSP_{LP}. The ATSP_{LP} provides a lower bound for the ATSP, which is useful for branch and bound solution techniques as well as for providing a proof of the quality of heuristic solutions. The integrality gap gives a measure of the "goodness" of this lower bound; moreover, a constructive proof showing this integrality gap is at most some constant α would provide an α-approximation algorithm for the ATSP. It is not known whether the integrality gap for the ATSP and the ATSP_{LP} is even bounded. This contrasts sharply with the symmetric case with metric costs where a $\frac{3}{2}$-approximation algorithm is known and the integrality gap has been shown to be at most $3/2$. It was conjectured that the integrality gap for the asymmetric case was $4/3$. In this thesis, we refute this claim. We compute the integrality gap exactly for $4 \leq n \leq 7$ and provide some partial results for $8 \leq n \leq 15$. We also provide two infinite families of extreme points of the polytope associated with the ATSP_{LP}, the Asymmetric Subtour Elimination Polytope (ASEP), whose integrality gaps approach $3/2$ and 2 respectively. Throughout this thesis, we explore several different properties of the extreme points of the ASEP. We provide two operations which can be used to create new extreme points. We also present an algorithm that can generate almost all of the extreme points of the ASEP for $4 \leq n \leq 7$ much more quickly than other polyhedral extreme point enumeration algorithms. We explore the facets of the ATSP-polytope and use them to further investigate the integrality gap. We finish this thesis with an exploration of the Strongly Connected Spanning Subgraph Problem (a close relative of the ATSP) and explore how much we can relax the ATSP and still have a hope of having a lower bound on the optimal value.

Acknowledgements

I am very grateful to my supervisor Sylvia for all her hard work in editing my papers, her help in dealing with university bureaucracy, her friendship, and her encouragement in my research. I am also thankful for Mateja's support and always-open door. Thank you to Karen, Pedro, Bob, Jeff, and Marwan for their friendship and for making University of Ottawa a fun place to be. Thanks too to Sebastian, Geneviève, Marcel, Marc, and Robert for their help with my various computer woes. Lastly, thank you to the teachers, administrators, and many other friends who brightened my days.

Dedication

To Carolyn and all my family. Thank you for all your support and encouragement.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introducing the Asymmetric Travelling Salesman Problem

The *Asymmetric Travelling Salesman Problem* (henceforth called the ATSP) is a famous problem in the field of Combinatorial Optimization and is the main subject of this thesis. The ATSP can be stated as follows. Given n cities and the cost of travelling from each city to every other, find a minimum cost itinerary such that

- all n cities are visited,
- every city is visited exactly once, and
- the itinerary ends with the same city with which it started.

We call any itinerary which meets the above criteria a tour. The significance of the word "asymmetric" has to do with the costs in that the cost of travelling from city A to city B may be different from the cost of travelling from city B to city A. This aspect of the problem distinguishes it from the more familiar *Symmetric Travelling Salesman Problem* (STSP) also called the *Travelling Salesman Problem* (TSP) in which the cost of travelling between two cities is the same, regardless of the direction.

The book "The Traveling Salesman Problem" [58] is an excellent resource on the STSP and the ATSP and contains a good overview of the history of developments in the study of these problems. Schrijver [76] also wrote about the early history of the STSP and ATSP. Throughout their history, much more attention has been focused on the STSP than on the ATSP. More real-world problems are modeled using the STSP as opposed to the ATSP. As well, the STSP is a simpler problem in the sense that it is the special case of the ATSP when the cost of travelling from city A to city B is the same as the cost of travelling from city B to city A.

The decision problem associated with the ATSP is known to be an NP-complete problem [53] which means that we do not know of any way of efficiently solving it in all cases and it is thought to be unlikely that such a method exists. This causes a problem, as there are many large-scale real world applications of this problem that require solutions.

Apart from the obvious application of the ATSP in solving routing problems such as vehicle routing problems and collecting coins from a set of pay phones, the ATSP has also been used to solve crew scheduling and industrial scheduling problems. The ATSP has been used to order the execution of parts of a computer program on a pipeline microprocessor to minimize wait-times [83], schedule stacker cranes at automatic warehouses [2], find the best colour sequence in the process of fabric dyeing to minimize the effects of earlier colours on subsequent ones [61], optimally insert a sequence of electronic components on a printed circuit board [59], and, interestingly, order the border pieces in an algorithm for solving jigsaw puzzles [41].

For the ATSP as for other NP-hard problems, two different approaches are taken when trying to obtain solutions – exact methods and heuristic methods. Exact methods are used to find exact solutions. These methods run in exponential time in general but they may solve a particular instance of interest in a reasonable amount of time. Heuristic methods are used to quickly find solutions which are not necessarily optimal. In general, there is no guarantee that heuristic methods find exact solutions, but they may frequently find exact solutions in practice.

1.1.1 Exact Methods

The earliest methods used to find exact solutions to integer programs (problems of minimizing linear functions over a set of linear constraints and variables with integer values) were cutting plane methods. The general framework was first presented and applied (to the STSP coincidentally) in Dantzig, Fulkerson, and Johnson's [27] seminal paper. The general idea of the cutting plane method is to solve a linear programming relaxation (allowing the values of the variables of the integer program to be real instead of just integer) of the integer program. If the optimal solution is not integer, then we add a valid inequality for the integer program which is violated by this solution to the linear program and re-solve. Gomory [42] refined this idea by a rounding technique in the following way. We start with our linear programming relaxation, written in equality form (by adding slack variables if necessary) and we find an optimal solution. If the solution is integral then it is optimal for the integer program. Otherwise, we use the fractional solution to create a new equality which is a linear combination of the constraints of the linear program. We then subtract the floor of each coefficient from itself to get a new valid inequality whose coefficients are each in the interval $[0, 1)$. We add a slack variable to this inequality and add the resulting to the linear program and continue. Later Chvátal [23] expanded on the theory of Gomory's ideas.

The next method used to exactly solve integer programs was the branch and bound method pioneered by Eastman [28] and Land and Doig [57]. Little, Murty, Sweeney, and Karel [60] applied this method specifically to the TSP. In this method, we break up the feasible region into subregions (branching) and find an upper and lower bound on the optimal value over each subregion (bounding). For minimization problems, if the upper bound of a subregion is less than the lower bound of another subregion then we prune the latter branch. For the ATSP, the lower bounds were originally obtained [28, 60] by optimizing over the assignment polytope. Murty [69] and Bellmore and Malone [7] later used violations of the subtour elimination constraints to branch.

The above two methods were merged by Padberg and Rinaldi [72] into the branch and cut method. Rather than finding Gomory cuts, instead they used a method of finding violated cuts called the template paradigm which was first suggested by Chvátal [24] with regards to using comb inequalities. In general, a template is a set of valid inequalities of a certain type (such as the subtour elimination constraints, blossom inequalities, comb inequalities, or clique tree inequalities). In the template paradigm, the template is also accompanied by an efficient separation algorithm, that is an algorithm, which, given a point which is not a solution to the integer program, attempts to find an inequality in the template violated by this point. In the branch and cut method, if we find a violated inequality, then we add it to our linear program and re-solve. We repeat this process until no violated inequality for the point can be found by the separation algorithm (although one may exist if the separation algorithm is not exact). If the current point is not integer, then we branch as in the branch and bound method.

This approach was applied to the ATSP by Fischetti and Toth [33] along with several new separation algorithms for some of the most famous facet-inducing inequalities of the ATSP polytope. This hybrid method was shown in their paper to be significantly faster and more effective in practice than the branch and bound methods of the time.

1.1.2 Heuristic Methods

Many researchers have also worked on developing heuristic methods to solve difficult combinatorial optimization problems – methods that work quickly in practice but do not guarantee exact solutions. In general, the most sought-after types of heuristics are α-approximation algorithms for some constant α. These are heuristics which find a solution in polynomial time and guarantee that the total cost of any solution generated by the heuristic is within a factor of α of the optimal value of an exact solution to the problem.

One of the earliest heuristics used for the ATSP was the nearest neighbour algorithm. This method was adopted from analogous methods for solving the STSP and runs in $O(n^2)$ time. For the nearest neighbour, we build a tour by starting with some partial itinerary ending at city A and we find a city B not already in the partial itinerary such that the cost of travelling from A to B is as cheap as possible. Once we have visited all the cities, we return from whence we came. Unfortunately, Frieze, Galbiati, and Maffioli [35] show that we can construct travelling costs among n cities such that a solution found by the nearest neighbour algorithm can be off by as much as a factor of roughly $n/2$ from the cheapest tour.

Another heuristic in use is the k-opt heuristic where $k \geq 3$. Here we take k partial itineraries which partition all the cities we would like to visit. We then consider adding roads in all possible ways that would augment these partial itineraries into a full tour. Again, Frieze, Galbiati, and Maffioli [35] show that, we can construct travelling costs among n cities such that a solution found by the k-opt heuristic can be off by as much as a factor which is of order n from the cheapest tour. Frieze, Galbiati, and Maffioli [35] presented their own algorithm which repeatedly optimizes over the assignment polytope and then merges the solutions into a tour. They show that the solution found by their algorithm (which runs in $O(n^3)$ time, as does 3-opt) can be off by a factor of at most $\lceil \log_2(n) \rceil$ from the cheapest tour.

More recently, Karp and Steele [54] presented a patching heuristic for the ATSP which has a runtime of $O(n^3)$. They would find a solution to the assignment problem and repeatedly patch together the largest subtours until they arrive at a full tour. In their analysis, they consider only independent randomly generated costs between 0 and 1 and show that their heuristic finds a solution with high probability that is within a constant factor of the optimal solution.

Johnson, Gutin, McGeoch, Yeo, Zhang and Zverovitch [51] provide much more information on heuristics for the ATSP.

1.1.3 Relaxations and Bounds

A question arises, however, when considering general integer programming problems. We use heuristics when it is not practical to solve the integer programming problem exactly and so how do we know whether or not a solution obtained as the output of a heuristic is close to optimal if it is impractical to solve the integer program exactly? For a minimization problem, the answer lies in finding a good lower bound on the optimal value of the integer program. If we relax the integrality constraint in the integer program, we get a linear program whose optimal value is a lower bound on the optimal value of the integer program. A natural question is to wonder how good is such a lower bound. A measure of this is the integrality gap of the problem which is the quotient of the optimal value of the integer program over the optimal value of the linear program. Notice that any proof that shows that the integrality gap of a given integer program is at most α, and gives a way to construct such an integer solution in polynomial time, provides an α-approximation algorithm for the problem.

For the ATSP, we use a well-known integer programming formulation and we call the resulting linear programming relaxation the ATSP_{LP}. The study of the integrality gap of this problem when the costs obey the triangle inequalities (i.e. the costs are said to be metric) is the main focus of this thesis. We show in Proposition 4.2.1 that, without this metric restriction, there is no upper bound on the value of the integrality gap. Worse yet, if we do not restrict our costs to obey the triangle inequalities, Sahni and Gonzalez [74] show that there can be no polynomial-time α-approximation algorithm for the ATSP (and also the STSP) unless $P = NP$. Carr and Vempala conjectured [16] in 2004 that the integrality gap of the problem when the costs obey the triangle inequality was 4/3. On the other hand, Williamson [81] shows that the integrality gap is at most $\lceil \log_2(n) \rceil$ but no constant (which does not depend on n) upper bound is known for this integrality gap. Nor is there any known α-approximation algorithm for the ATSP for any constant α. This contrasts sharply with the analogous integrality gap for the STSP which was proved to be at most 3/2 [82, 77], and is known to be at least 4/3 in the worst-case (and a folklore conjecture states that this is also the upper bound). Better yet, there is a well-known 3/2-approximation algorithm [22] for the STSP.

In this thesis, we calculate the exact integrality gap of the ATSP for small values of n. In order to accomplish this in a reasonable amount of time, we fix the optimal solution of the ATSP_{LP} and calculate the maximum integrality gap relative to that solution. Specifically, we study the extreme points of the feasible region of the ATSP_{LP} (which, in fact, define the entire feasible region) to learn many structural properties of these points and exploit them in our calculation of the integrality gap.

Our main contributions in this thesis to a better understanding of the integrality gap of the ATSP are as follows. Most importantly, we present two simple structural operations which can be applied to known extreme points to obtain new ones and we show that one of these operations never increases the resulting integrality gap. We apply these operations to exactly compute the integrality gap of the ATSP for $4 \leq n \leq 7$ and we provide a lower bound on the integrality gap for $8 \leq n \leq 15$. Next, we discover two families of costs which show that the integrality gap is at least 3/2 and 2 respectively. This family of costs that proves that the integrality gap is at least 3/2 was discovered independently, but at the same time as the family presented by Charikar, Goemans, and Karloff [18] which shows that the integrality gap is at least 2. Lastly, we provide a heuristic for quickly generating the extreme points of the feasible region of the ATSP_{LP} for small values of n.

In the next section, we provide a more detailed outline of the contents of the thesis.

1.2 Thesis Outline

Throughout this thesis, we study the integrality gap of the ATSP, paying special attention to a linear programming relaxation of this problem whose feasible region is the Asymmetric Subtour Elimination Polytope (ASEP). We see that these results differ significantly from their undirected analogues in the STSP and the Subtour Elimination Polytope.

In the second chapter, we introduce the main notation and terminology that is used in this thesis. We also present the basics of the main tools that we use to study the ATSP and the ASEP - namely graph theory, polyhedral theory, and linear programming.

In the third chapter, we explore various properties of the ASEP. Specifically, we investigate certain properties of the extreme points of the ASEP having to do with the structure of their cobases. We present a new operation, called "inserting a 1-arc" which allows us to create new extreme points of the ASEP from those that are already known. We introduce a new operation called a "2-jack" that is used to create new half-integer extreme points of the ASEP from those already known.

In the fourth chapter, we introduce a measure that is a key theme for the rest of the thesis - the integrality gap. The integrality gap is the maximum (over all cost functions for a given n) ratio of the optimal value of the ATSP to the optimal value over the ASEP. As with many other authors, we restrict our attention to arc-costs that obey the triangle inequalities. We present new results by computing the integrality gap exactly for $4 \leq n \leq 7$. We also provide a lower bound on the value of the integrality gap for $8 \leq n \leq 9$. The lower bound on the integrality gap attained for $n = 9$ was $\frac{11}{8}$ - a value never attained by any known example of the undirected analogue of this measure! Furthermore, this refutes a conjecture by Carr and Vempala [16] that the integrality gap is at most $\frac{4}{3}$. We finish the chapter by demonstrating the usefulness of the 1-arc insertion operation in that it never increases the integrality gap.

In the fifth chapter of this thesis, we present two families of ASEP extreme points that yield relatively large integrality gaps. The first of these families is obtained by observing a pattern among the extreme points which attained the maximal integrality gap in the previous chapter. Specifically, we repeatedly apply the 2-jack operation to the half-integer extreme points which attain the largest integrality gaps for $4 \leq n \leq 9$. As a result, we discover that the integrality gaps of the members of this family can be made arbitrarily close to $\frac{3}{2}$. In the second part of this chapter, we recursively construct another family of half-integer extreme points which can attain an integrality gap arbitrarily close to 2. This value surpasses the well-known result that the integrality gap of the undirected analogue is at most $\frac{3}{2}$. We finish the chapter by comparing our family to another one presented by Charikar, Goemans, and Karloff [18].

In the sixth chapter, we turn the problem around and consider computing the integrality gap starting with some well-known cost functions. We consider some famous facet-inducing inequalities of the ATSP. For each facet-inducing inequality, we find an equivalent metric version that maximizes the integrality gap. We present these new results and discover that the largest integrality gap we find by these methods is $\frac{6}{5}$.

In the seventh chapter, we explore the cobases of the extreme points of the ASEP. We discover from our complete lists of extreme points for $4 \leq n \leq 7$ that the majority of extreme points of the ASEP have cobases which can be considered "unique" in a certain sense. We then present an algorithm which attempts to map out the polytope by pivoting from one extreme point to the next. Due to the "uniqueness" of the cobases, we avoid degeneracy entirely. We also avoid pivoting from extra isomorphic copies of a given extreme point. As a result, we end up with an algorithm that can find almost all of the extreme points of the ASEP for small values of n in an amount of time that is significantly less that the exhaustive methods.

In the eighth chapter, we consider a problem that is closely related to the ATSP - namely the Strongly Connected Spanning Subgraph Problem (SCSSP for short).

We present an interesting result about the structure of members of a sufficient list of solutions which are optimal with respect to cost functions that obey the triangle inequalities. This is the directed analogue of a result by Monma, Munson, and Pulleyblank [66] and is useful for computing the exact integrality gap of the SCSSP for small values of n. We also show that the ratio of the optimal value of the ATSP to the optimal value of the SCSSP is proportional to n which contrasts sharply with the undirected analogue of this measure and its proven upper bound of $\frac{4}{3}$.

In the ninth chapter, we present our conclusions and suggest further work.

Chapter 2

Preliminaries and Notation

In this chapter, we present the basic terminology and theoretical results which are used in this thesis. We begin with a discussion of the graph theory notation that is used. In the second section, we add extra information to our directed graphs and encode this information using vectors. This vector approach is also used in the third section as we present some relevant ideas from polyhedral theory. We finish off by merging all these ideas into the basics of linear optimization.

2.1 Graphs and Digraphs

In this section we present the basic notation and terminology we use when discussing directed graphs. For further information, an excellent book that thoroughly introduces graph theory is Bondy and Murty's "Graph theory with applications" [11].

A *directed graph* (or more simply *digraph*), $G = (V, E)$, is an ordered pair of a finite set, V, and a multi-set, E. The elements of V are called *nodes* and the elements of E are called *arcs.* Each arc is an ordered pair of nodes. Formally, if $e \in E$ such that $e = (u, v)$ where $u, v \in V$ then we simplify the notation to $e = uv$. We call u and v the *endpoints* of e and we say that e is *incident* to u (or v). Furthermore, u is called the *tail* of e and v is the *head* of e. If two distinct arcs $e, f \in E$ have the same tail and the same head then we say that these arcs are *parallel.* If, on the other hand, two arcs of G have the same endpoints but are not parallel then we say that these arcs are *anti-parallel* or that they form a *digon.* If a digraph has parallel arcs then we call it a *directed multigraph* or a *multi-digraph.* For the purposes of this thesis, no directed graph or directed multigraph has an arc $uv \in E$ such that $u = v$. Usually we let n denote the number of nodes in a digraph or multi-digraph. As well, K_n denotes the *complete digraph* which has n nodes and a single arc for each ordered pair of (distinct) nodes.

Digraphs and multi-digraphs are often represented pictorially with dots representing the nodes and arrows representing the arcs where the arrows point from the tail to the head. An example of the picture of a multi-digraph is shown in figure 2.1. In this multi-digraph, we have labelled the nodes $V = \{1, 2, 3, 4, 5\}$ so $E = \{12, 14, 15, 21, 23, 31, 43, 43, 52\}$.

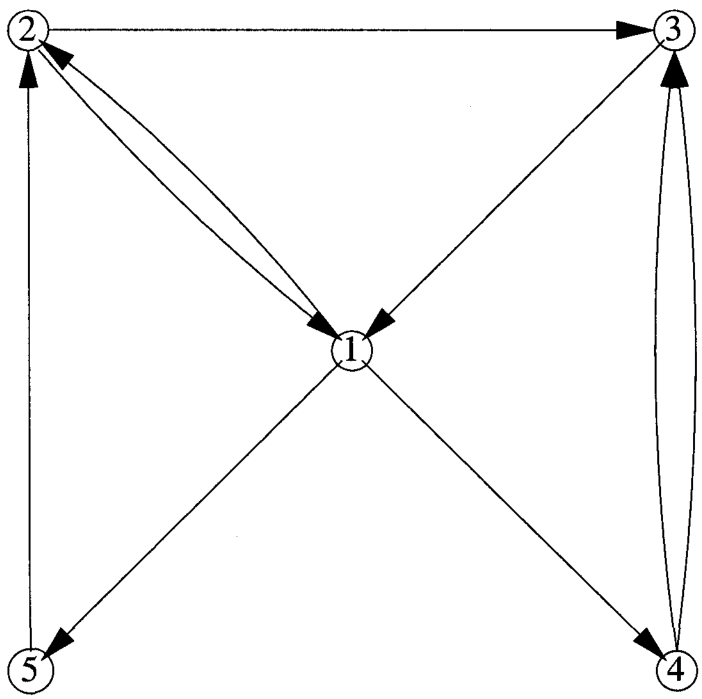

Figure 2.1: A picture of a multi-digraph

For $u, v \in V$, if u and v are the endpoints of some arc in E then we say that u

and v are *adjacent* or that they are *neighbours.* For a given node $v \in V$ the *outdegree* of v is the number of arcs of E that have v as their tail whereas the *indegree* of v is the number of arcs of E that have v as their head. For example, in Figure 2.1, the neighbours of node 4 are the nodes 3 and 1 respectively. Also node 4 has outdegree 2 and indegree 1.

For any $\emptyset \subset S \subset V$ we denote

$$\delta_G(S) = \{uv \in E \mid u \in S,\ v \in V \backslash S\}$$

and we call $\delta(S)$ the *cut of G induced by S.* For notational simplicity, we define $\delta_G(v) = \delta_G(\{v\})$ for any $v \in V$ and we drop the subscript G whenever the digraph or multi-digraph being referenced is obvious. For any $\emptyset \subset S \subseteq V$ we denote

$$\gamma_G(S) = \{uv \in E \mid u \in S,\ v \in S\}.$$

Again for notational simplicity, we drop the G subscript whenever the digraph or multi-digraph being referenced is obvious. Also for any $\emptyset \subset S \subseteq V$ we denote

$$G[S] = (S, \gamma_G(S))$$

which we call the *subgraph of G induced by S* and is the directed graph or multigraph we obtain by removing all the nodes of G which are not in S and all arcs incident to these nodes. In Figure 2.2 we show the subgraph of the multi-digraph shown in Figure 2.1 induced by $\{1, 2, 3\}$. In general, for any $\emptyset \subset S \subseteq V$ and $F \subseteq \gamma_G(S)$, the digraph (or multi-digraph) (S, F) is called a *subgraph* of G.

For any $F \subseteq E$ we define $G - F = (V, E \backslash F)$, that is the digraph or multi-digraph obtained by *deleting the arcs* of F. If F consists of a single arc $e = uv$ then we usually simplify the notation to $G - e$ or $G - uv$. Given some $\emptyset \subset S \subset V$ we define the operation of *deleting the nodes* of S from G and denote this operation by $G - S = G[V \backslash S]$. Again, if S consists of a single node v then we usually simplify the notation to $G - v$. For any $\emptyset \subset S \subset V$ we define the digraph or multi-digraph obtained by *identifying the nodes of S* as the digraph or multi-digraph obtained by

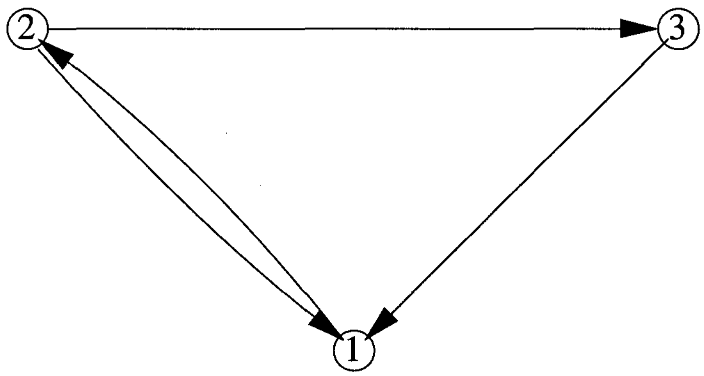

Figure 2.2: Example of an induced subgraph

deleting the nodes of S from G, adding a new node, call it z, and adding arcs in the following way: if $uv \in \delta_G(V \backslash S)$ then uz is an arc of the new digraph or multi-digraph and if $uv \in \delta_G(S)$ then zv is an arc of the new digraph or multi-digraph. For the special case when S contains precisely the endpoints of an arc $e = uv$ of G then we call this *contracting the arc* and we denote the resulting digraph or multi-digraph by G/e or G/uv. An example of contracting the arc 23 of the multi-digraph shown in Figure 2.1 is shown in Figure 2.3 where we have labelled the new node created with the label 6.

A sequence of the arcs of a digraph of multi-digraph $u_1v_1, u_2v_2, \ldots, u_kv_k$ where $v_i = u_{i+1}$ for each $1 \leq i \leq k-1$ is called a *walk*. As such, we usually represent a walk by the ordered sequence of nodes $(u_1, u_2, \ldots, u_k, v_k)$. If $v_k = u_1$ then we call it a *closed walk*. A walk which does not contain the same arc more than once is called a *trail* and a closed walk with this property is called a *closed trail*. If a digraph or multigraph G has the property that there exists a closed trail of G that contains each arc of G exactly once then G is said to be *Eulerian*.

A trail $(v_1, v_2, \ldots, v_k)$ such that v_1, v_2, $\ldots$, v_k are all distinct except perhaps $v_k = v_1$ is called a *dipath* or a *(v_1, v_k)-dipath* and we say that v_1, v_2, $\ldots$, v_k are *covered by the dipath*. If a dipath covers all the nodes of a given digraph or multi-digraph, then we call this dipath a *directed Hamilton path*. The nodes v_2, $\ldots$, v_{k-1}

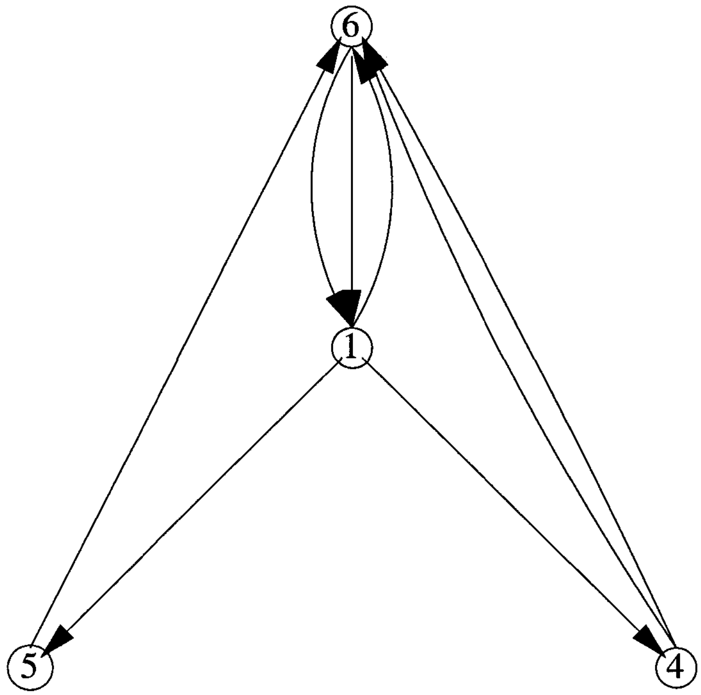

Figure 2.3: Example of contracting an arc

are called the *internal nodes* of the dipath. If $v_k = v_1$, then we call this dipath a *dicycle.* If G contains a dicycle which covers all the nodes of G then G is said to be *hamiltonian* and this dicycle is called a *tour* or a *directed Hamilton cycle.* Two dipaths or dicycles are said to be *arc-disjoint* if they do not share a common arc. Similarly two dipaths are said to be *internally node-disjoint* if they do not share any common internal node. Finally, two dicycles are said to be *node-disjoint* if they do not cover any common node. A dicycle or dipath is said to have *length* k if it contains precisely k arcs. Such a dicycle or dipath is sometimes referred to as a k-dicycle or k-dipath respectively. A *shortest* (u, v)*-dipath* in a multi-digraph or digraph G is a minimum-length (u, v)-dipath. Notice that such a dipath may not be unique.

If a digraph or multi-digraph $G = (V, E)$ has the property that there exists a (u, v)-dipath in G for each ordered pair of distinct nodes $u, v \in V$ then we say that G is *strongly connected.*

For the purposes of contrast, we often compare our results for digraphs to those of undirected graphs. A *graph* or *undirected graph*, $G = (V, E)$, is an ordered pair of finite sets. The elements of V are called *vertices* and the elements of E are called *edges*. Each edge is an unordered pair of vertices. Essentially, a graph is a digraph where we ignore the directions of the arcs. Basically, all of the above definitions for digraphs or multi-digraphs can be restated for undirected graphs (or multigraphs) by eliminating the ordering of the endpoints in any arc. In terms of the terminology, we just drop the word "directed" or the prefix "di-". Specifically, we lose the notion of heads, tails, and anti-parallel arcs and we simply talk of the *degree* of a vertex instead of the indegree and outdegree of a node.

We also call a graph $G = (V, E)$ *connected* if there is a $\{u, v\}$-path in G for every pair of distinct vertices $u, v \in V$ (as opposed to its directed analogue which is strongly connected). A graph that is connected but has no cycles is called a *tree*. If H is a subgraph of G which covers all the vertices of G then we say that H *spans* G. If G has a tree as a subgraph that spans G then we call such a tree a *spanning tree*. We also say that G is *2-vertex-connected* if $G - v$ is connected for all $v \in V$.

2.2 Vectors, Digraphs, and Arc-values

In this section, we introduce extra layers of information to the complete digraph and we present notation we use to talk about this extra data.

Given the complete digraph $K_n = (V, E)$ on n nodes, consider associating a real value c_e with each arc $e \in E$. We sometimes write $c \in \mathbb{R}^E$ as a compact form. We refer to c_e as the *arc-cost* of e and as such we call c a *cost function* or a *cost vector*. An example of K_4 with a cost function is depicted in Figure 2.4. For any $F \subseteq E$ we define $c(F) = \sum_{e \in F} c_e$. For simplicity, if we have a certain subgraph $G' = (V', E')$ of K_n then we call $c(E')$ the *cost of G' with respect to c*. Hence we can talk about the cost of a dipath, a dicycle, or a tour with respect to c. More importantly, we can discuss a minimum cost dipath, dicycle, or tour of K_n with respect to c. When the cost function is clear, we often omit "with respect to c". An example of a minimum cost tour of the digraph with the cost function shown in Figure 2.4 is given in Figure 2.5.

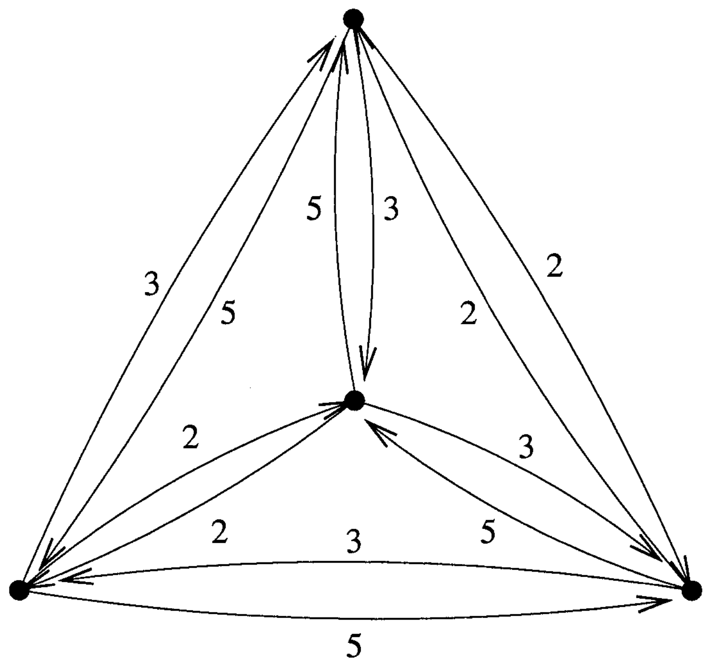

Figure 2.4: K_4 with arc-costs

Of course, there is no need to ascribe the term "cost" to values associated to the arcs of K_n. For any $x \in \mathbb{R}^E$ call x_e the *x-value* of e. As a special case, if $x_e = 1$ for some $e \in E$ then we say that e is a *1-arc*. For any $F \subseteq E$ we still use the notation $x(F) = \sum_{e \in F} x_e$. Appealing to the language of network flows, we sometimes refer to x_e as the *flow* on the arc $e = uv$. If $x_{uv} > 0$ and P is some (u,v)-dipath in K_n, then we can construct a new flow on the arcs of E by increasing the flow on all arcs of P by x_{uv} and then reducing the flow on uv to 0. We call this operation *re-routing the flow on uv through P*.

Lastly, we might assign both x-values and costs to the arcs of K_n. In this case we use the notation that $cx = \sum_{e \in E} c_e x_e$. For a given subgraph $G' = (V', E')$ of K_n

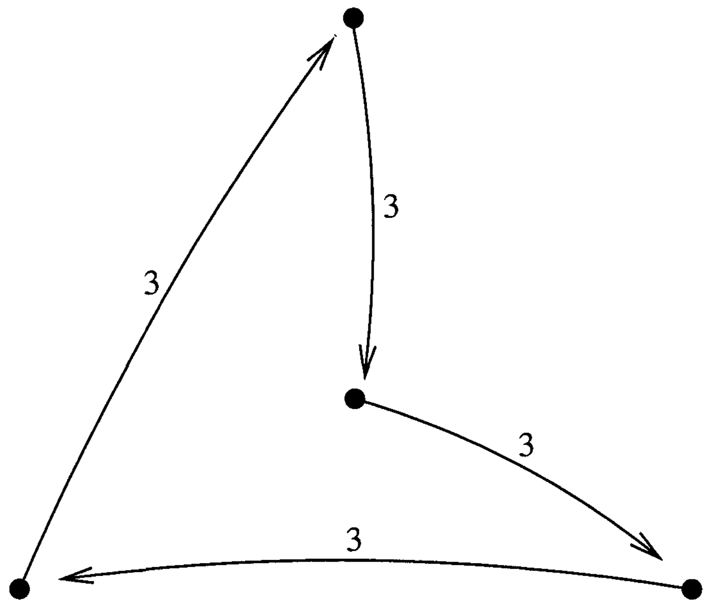

Figure 2.5: A minimum cost tour

we define the *characteristic vector of* G' as $x \in \mathbb{R}^E$ where

$$x_e = \begin{cases} 1 & \text{if } e \in E' \\ 0 & \text{otherwise.} \end{cases}$$

Since the characteristic vector depends only on the arcs in the subgraph, we can extend our definition in the obvious way to encompass objects such as dipaths, dicycles, and tours. For example, if c is the cost function on K_4 as shown in Figure 2.4 and x is the characteristic vector of the tour depicted in Figure 2.5 then $cx = 12$ (which is the total cost of the tour).

We use this extra information and vector notation to pose various optimization problems for digraphs. We use this terminology in the next section to present the basics of polyhedral theory.

2.3 Polyhedral Theory

A *polyhedron* is a subset of $\mathbb{R}^d$ defined as the set of points which satisfy a finite number of linear inequalities. A *polytope* is a bounded polyhedron – that is a polyhedron $\mathcal{P} \subset \mathbb{R}^d$ such that there exists some $B > 0$ where $\mathcal{P} \subseteq \{x \in \mathbb{R}^d \mid \|x\| \leq B\}$. All of the results in this section are well known and can be found in any text on polyhedral theory such as *Integer and Combinatorial Optimization* by Nemhauser and Wolsey [71].

Let $\mathcal{P}$ be a polyhedron and let $x \in \mathcal{P}$. We say that x is an *extreme point* of $\mathcal{P}$ if the only points $y, z \in \mathcal{P}$ which satisfy $x = \frac{1}{2}y + \frac{1}{2}z$ are $y = z = x$. A *convex combination* of a set of points $\{x^1, \ldots, x^k\} \subset \mathbb{R}^d$ is a linear combination $\alpha_1 x^1 + \ldots + \alpha_k x^k$ whose coefficients satisfy the following properties.

$$\sum_{i=1}^{k} \alpha_i = 1$$

$$\alpha_i \geq 0 \text{ for each } 1 \leq i \leq k.$$

The *convex hull* of a set of points $\{x^1, \ldots, x^k\} \subset \mathbb{R}^d$ is the set of all convex combinations of these points. We say that a polyhedron is *pointed* if it contains at least one extreme point. A consequence of Minkowski's Theorem [65] gives us the following description of a polytope.

Theorem 2.3.1. *A pointed polytope is equal to the convex hull of its extreme points.*

Hence, we see that we can completely describe a polytope by either giving a finite number of linear inequalities which define the polytope or by giving a list of all the extreme points of the polytope. In the special case that all the extreme points of a polytope consist of only integer values, we call the polytope *integral.*

More generally, the extreme points are not usually sufficient to describe an arbitrary polyhedron. If we have a polyhedron $\mathcal{P}$ defined as all the solutions to the finite set of linear inequalities $Ax \geq b$ and we have some non-zero vector $r \in \mathbb{R}^d$ such that $Ar \geq 0$ then we call r a *ray* of $\mathcal{P}$. Furthermore, we say that r is an *extreme ray* of $\mathcal{P}$

if the only rays s and t of $\mathcal{P}$ which satisfy $r = \frac{1}{2}s + \frac{1}{2}t$ are positive scalar multiples of r.

The reason we are interested in the extreme rays of a polyhedron is that they, along with the extreme points, completely describe the polyhedron. The way in which this is accomplished is presented in Theorem 2.3.2.

Theorem 2.3.2. *Let $\mathcal{P}$ be a polyhedron with extreme points $x^1, \ldots, x^k$ and extreme rays $r^1, \ldots, r^l$. Then any $z \in \mathcal{P}$ can be expressed as*

$$z = \sum_{i=1}^{k} \alpha_i x^i + \sum_{j=1}^{l} \mu_j r^j$$

where $\alpha_i \geq 0$ for all $1 \leq i \leq k$, $\mu_j \geq 0$ for all $1 \leq j \leq l$, and $\sum_{i=1}^{k} \alpha_i = 1$.

The extreme point and extreme ray description of a polyhedron is particularly useful when we are doing optimization (see the next section) over the polyhedron.

Proposition 2.3.3. *Let $\mathcal{P} \subseteq \mathbb{R}^d$ be a pointed polyhedron and let $c \in \mathbb{R}^d$. If $\min\{cx \mid x \in \mathcal{P}\}$ is finite then it has an optimal solution which is an extreme point. Otherwise, there is an extreme ray r of $\mathcal{P}$ such that $cr < 0$ and hence $\min\{cx \mid x \in \mathcal{P}\}$ is unbounded.*

Throughout this thesis, we are interested in finding all the extreme points of a certain polytope. We often generate these extreme points using the following alternate definition.

Theorem 2.3.4. *Let $\mathcal{P}$ be a polytope and let $x \in \mathcal{P}$. Then x is an extreme point of $\mathcal{P}$ if and only if x is the unique solution to a system of equations obtained from a subset of the inequalities which define $\mathcal{P}$ set to equality.*

For a given extreme point $x \in \mathcal{P} \subseteq \mathbb{R}^d$, consider the subset of the inequalities which define $\mathcal{P}$ which hold with equality for x. If we change all the inequality signs to equality in this subset, we get a system of linear equations which has x as its unique solution. Any subsystem with exactly d equations which has the unique solution x

is called a *cobasis* of x. Two distinct extreme points x and y are called *adjacent* if there are cobases for x and y which have precisely $d-1$ equations in common.

Next we introduce a measure of the polyhedron which leads to yet another important way of describing the polyhedron.

A set of points $\{x^1, \ldots, x^k\} \subset \mathbb{R}^d$ is said to be *affinely independent* if the only solution to the pair of equations

$$\begin{aligned} \sum_{i=1}^{k} \alpha_i x^i &= 0 \\ \sum_{i=1}^{k} \alpha_i &= 0 \end{aligned}$$

is $\alpha_i = 0$ for each $1 \leq i \leq k$. Notice that every linearly independent set of points is also affinely independent but the converse is not true. However, if $\{x^1, \ldots, x^k\}$ are affinely independent then the set of points $\{x^2 - x^1, \ldots, x^k - x^1\}$ must be linearly independent. The *dimension* of $\mathcal{P}$, which we denote $\dim(\mathcal{P})$, is one less than the maximum number of affinely independent points in $\mathcal{P}$.

We can use this measure of dimension to give another way to completely describe a polyhedron. First consider an inequality $\alpha x \geq \alpha_0$ which is satisfied by all points x in some polyhedron $\mathcal{P}$. We call such an inequality a *valid inequality.* Now consider $F = \{x \in \mathcal{P} \mid \alpha x = \alpha_0\}$ which is itself a polyhedron. If $\dim(F) = \dim(\mathcal{P}) - 1$ then we say that F is a *facet* of $\mathcal{P}$ and we call $\alpha x \geq \alpha_0$ a *facet-inducing inequality.*

Now let $\mathcal{P} = \{x \in \mathbb{R}^d \mid Ax \geq b\}$ be a polyhedron. Let $a_1 x \geq b_1, \ldots, a_k x \geq b_k$ be the inequalities of $Ax \geq b$ which actually hold with equality for all $x \in \mathcal{P}$. Thus $a_i x = b_i$ for all $x \in \mathcal{P}$ and for each $1 \leq i \leq k$ and so we call this set of inequalities the *equality set of* $\mathcal{P}$. We say that two valid inequalities $\alpha x \geq \alpha_0$ and $\beta x \geq \beta_0$ are *equivalent* if there exist $\mu_1, \ldots, \mu_k \in \mathbb{R}$ and $\lambda > 0$ such that

$$\beta = \lambda \alpha + \sum_{i=1}^{k} \mu_i a_i$$

and

$$\beta_0 = \lambda \alpha_0 + \sum_{i=1}^{k} \mu_i b_i.$$

A well-known theorem in polyhedral theory states the following.

Theorem 2.3.5. *Two facet-inducing inequalities are equivalent if and only if they induce the same facet.*

We can then use facet-inducing inequalities to define a polyhedron as follows.

Theorem 2.3.6. *Let $\mathcal{P}$ be a polyhedron. The equality set of $\mathcal{P}$ along with a single facet-inducing inequality corresponding to each facet of $\mathcal{P}$ is sufficient to describe the polyhedron.*

Thus we have our second way of describing the polyhedron. We use both the extreme point description and facet description to describe the polytopes that are important in this thesis.

2.4 Integer and Linear Programming

A *linear program* is a problem of minimizing (or maximizing) a linear function over a polyhedron in $\mathbb{R}^d$ defined by a finite set of linear inequalities and equalities which we call *constraints.* For simplicity in this section, we discuss minimization problems although all the terminology and results apply to maximization problems as well. Since every linear equality $\alpha x = \alpha_0$ can be replaced by the inequalities $\alpha x \geq \alpha_0$ and $\alpha x \leq \alpha_0$ and any inequalities of the form $\alpha x \leq \alpha_0$ can be replaced by $-\alpha x \geq -\alpha_0$ without changing the set of solutions to our constraints, we can write any linear program in the form

$$\begin{array}{lrcl} \text{minimize} & cx & & \\ \text{subject to} & Ax & \geq & b \\ & x & \in & \mathbb{R}^d. \end{array}$$

An excellent resource on this subject is Schrijver's [75] book entitled "Theory of linear and integer programming".

We call the polyhedron $\{x \in \mathbb{R}^d \mid Ax \geq b\}$ the *feasible region* and any point in this polyhedron is called a *feasible point* or *feasible solution.* The formal sum cx, that is $\sum_{i=1}^{d} c_i x_i$ is called the *objective function* although we sometimes refer to c itself as the objective function for simplicity. For a given feasible solution x' we call the value cx' the *objective value* of x'. If $\min\{cx \mid Ax \geq b,\ x \in \mathbb{R}^d\}$ is finite then we call this quantity the *optimal value* of the linear program. Any feasible solution x' which has an objective value that is equal to the optimal value is called an *optimal solution.*

A common method for solving linear programs in practice is the Simplex Method which was developed by George Dantzig [26]. This method takes an extreme point of the feasible region, removes one of the constraints in its cobasis and adds another (a process called *pivoting*) to generate a new adjacent extreme point whose objective value is no worse that the first extreme point. For each extreme point, we create a potential dual solution (see below) and we terminate the algorithm with an optimal solution once this potential dual solution is actually feasible. Technically speaking, the simplex method is not a polynomial time algorithm as shown by Klee and Minty [56]. However, Borgwardt [12] shows that it does perform well on average. Later, Khachiyan [55] proves that linear programming problems can be solved in polynomial time using the Ellipsoid Method, but this method does not generally perform as well in practice as the Simplex Method. We present a similar method to the Simplex Method in Chapter 7 for finding many of the extreme points of an important polytope.

Consider a linear program with q constraints of the form

$$\begin{array}{lrcl} \text{minimize} & cx & & \\ \text{subject to} & Ax & \geq & b \\ & x & \in & \mathbb{R}^d. \end{array}$$

The *dual* of this linear program is

$$\begin{array}{lrcl} \text{maximize} & by & & \\ \text{subject to} & A^T y & = & c \\ & y & \geq & 0 \\ & y & \in & \mathbb{R}^q. \end{array}$$

We call the original linear program the *primal* in relation to the dual. The importance of the dual is summarized in the following two theorems called the *weak duality theorem* and the *strong duality theorem*, respectively.

Theorem 2.4.1. *If x' is any feasible solution to the primal and y' is any feasible solution to the dual then*

$$by' \leq cx'.$$

Theorem 2.4.2. *If the primal and dual problems both have finite optimal values then these optimal values are the same.*

Hence, the weak duality theorem tells us that the optimal value of the dual provides a lower bound on the optimal value of the primal. Furthermore, the strong duality theorem demonstrates that if we are only concerned with finding the optimal value of a linear programming problem then we can solve either the problem itself or its dual – whichever is easier.

Better yet, we can gain more information about the nature of the optimal solutions. Let $a_i x \geq b_i$ for each $1 \leq i \leq q$ be the inequalities of $Ax \geq b$. Let x' be a feasible solution of the primal and let y' be a feasible solution of the dual. The *complementary slackness conditions* are

$$\text{“if } y'_i > 0 \text{ then } a_i x' = b_i \text{ for each } 1 \leq i \leq q \text{ ”.}$$

This leads us to the following helpful theorem.

Theorem 2.4.3. *Points x' and y' are optimal solutions of the primal and dual respectively if and only if the complementary slackness conditions hold.*

Hence we have another way of determining whether or not we have an optimal solution. More importantly, these complementary slackness conditions show us a connection between the optimal solutions of the primal and the dual. We use these concepts of duality and complementary slackness in Chapter 4.

Lastly, an *integer linear program* or simply *integer program* is a linear program where the feasible region is restricted to $\mathbb{Z}^d$. That is, a problem of the form

$$\begin{array}{lrcl} \text{minimize} & cx & & \\ \text{subject to} & Ax & \geq & b \\ & x & \in & \mathbb{Z}^d. \end{array}$$

Note that, in general, finding solutions to integer programs is an NP-complete problem.

Throughout this thesis, we study a specific integer program – namely one that completely describes the ATSP. In the next chapter, we see how we can formulate the ATSP as an integer program. We also study a linear program that is closely related to the ATSP.

Chapter 3

Basic Properties of the Asymmetric Subtour Elimination Polytope

We introduce the basics of graph theory, linear programming, and polyhedral theory in the last chapter. In this chapter we bring together all these related areas of study to explore the Asymmetric Travelling Salesman Problem. We begin with an integer programming formulation of the ATSP. We then discuss a related polytope and explore some of the important properties of this polytope.

3.1 The Asymmetric Subtour Elimination Polytope

Given $K_n = (V, E)$, attempts to solve the ATSP with linear programming techniques began with optimizing the linear program

$$\begin{array}{lrcll} \text{minimize} & cx & & & \\ \text{subject to} & x(\delta(v)) & = & 1 & \text{for all } v \in V \\ & x(\delta(V \backslash \{v\})) & = & 1 & \text{for all } v \in V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

The feasible region of this linear program is called the *assignment polytope.* Clearly, the characteristic vector of any tour is a point in the assignment polytope so the above linear program is a relaxation of the ATSP. Birkhoff [9] shows that the assignment polytope is integral and so we can use linear programming techniques to solve the associated integer program. Unfortunately, each extreme point of the assignment polytope is the characteristic vector of a collection of node-disjoint dicycles that span K_n. If we use the above linear program to try to solve the ATSP and we get an optimal solution that is a characteristic vector of a single dicycle, then this is a tour and we have, in fact, found an optimal solution to the ATSP. If, however, the optimal solution over the assignment polytope is the characteristic vector of a collection of more than two node-disjoint dicycles then we have not found a feasible solution of the ATSP. As a result, we need to add more constraints to this optimization problem in order to guarantee that we find a solution to the ATSP that is both feasible and optimal.

In 1954, Dantzig, Fulkerson, and Johnson [27] presented another set of constraints that are valid for the Symmetric TSP and eliminate the characteristic vectors of collections of more than one vertex-disjoint cycles from the feasible solutions to the undirected analogue of the assignment polytope. These constraints are easily adopted to the directed context and yield the following integer programming formulation of the ATSP.

$$
\begin{array}{llcll}
\text{minimize} & cx & & & \\
\text{subject to} & x(\delta(v)) & = & 1 & \text{for all } v \in V \\
 & x(\delta(V\backslash\{v\})) & = & 1 & \text{for all } v \in V \\
 & x(\delta(S)) & \geq & 1 & \text{for all } \emptyset \subset S \subset V \text{ where } 2 \leq |S| \leq n-2 \\
 & x_e & \geq & 0 & \text{for all } e \in E \\
 & x & \in & \mathbb{Z}^E. &
\end{array}
$$

A characteristic vector, x, of a collection of more than one node-disjoint dicycles is not a feasible solution to this integer program because if S is the set of nodes covered by one of these dicycles then $x(\delta(S)) = 0$.

Let $\mathcal{P}_T^n$ be the convex hull of all the feasible solutions of the above integer program. We call $\mathcal{P}_T^n$ the ATSP-polytope. If we then proceed to relax the integrality constraint of the ATSP-polytope, we get the Asymmetric Subtour Elimination Polytope, which is strongly related to the ATSP and whose structure and properties are central to this thesis. Unfortunately, unlike the assignment polytope, the Asymmetric Subtour Elimination Polytope is not integral.

Let $K_n = (V, E)$ be the complete digraph on $n \geq 3$ nodes. The *Asymmetric Subtour Elimination Polytope* (henceforth called the ASEP) on n nodes is the set of all $x \in \mathbb{R}^E$ such that

$$x(\delta(v)) = 1 \quad \text{for each } v \in V \tag{3.1.1}$$

$$x(\delta(V\backslash\{v\})) = 1 \quad \text{for each } v \in V \tag{3.1.2}$$

$$x(\delta(S)) \geq 1 \quad \text{for each } \emptyset \subset S \subset V \text{ where } 2 \leq |S| \leq n-2 \tag{3.1.3}$$

$$x_e \geq 0 \quad \text{for each } e \in E. \tag{3.1.4}$$

We let $\mathcal{P}_S^n$ denote the ASEP on n nodes. We call the equalities (3.1.1) and (3.1.2) the *node equalities*, the constraints (3.1.3) are called the *subtour elimination constraints* (or alternatively the *cut constraints*), and the inequalities (3.1.4) are called the *nonnegativity constraints.*

Extreme points of $\mathcal{P}_S^3$, $\mathcal{P}_S^4$, and $\mathcal{P}_S^5$ are depicted in Figure 3.1, Figure 3.2, and Figure 3.3 respectively. In each figure, the number next to each arc e is its x-value x_e and for every arc not shown, the x-value is 0. We can verify that any one of these are indeed extreme points by checking that the inequalities which define the ASEP and hold with equality for the given point define a system of equations which has a unique solution.

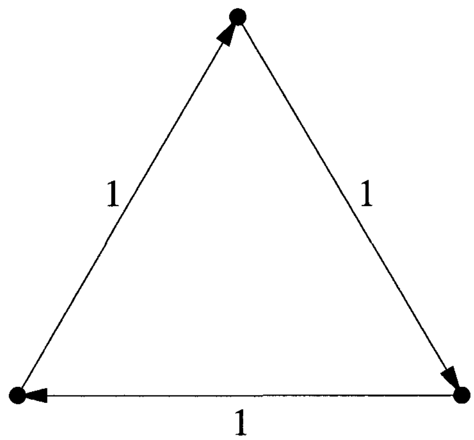

Figure 3.1: An extreme point of $\mathcal{P}_S^3$

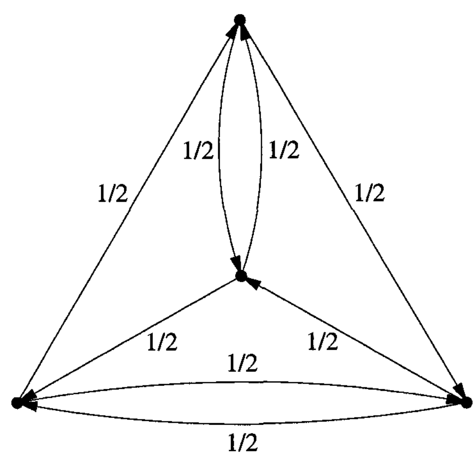

Figure 3.2: An extreme point of $\mathcal{P}_S^4$

As was mentionned before (and seen above), $\mathcal{P}_S^n$ is not integral. Therefore, we

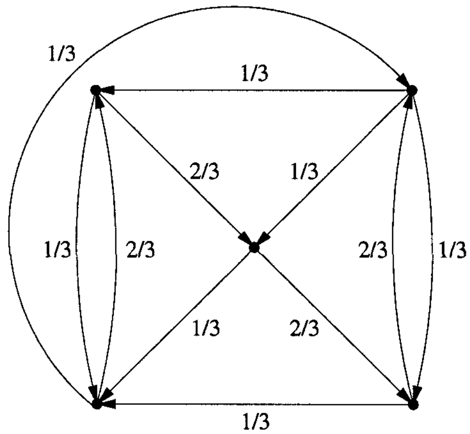

Figure 3.3: An extreme point of $\mathcal{P}_S^5$

cannot optimize over the ASEP and be guaranteed to find an optimal solution of the ATSP. However, optimizing over the ASEP does provide a lower bound on the optimal value of the ATSP since $\mathcal{P}_T^n \subseteq \mathcal{P}_S^n$. Discovering how good a lower bound this gives is the main focus of this thesis.

Consider again the assignment polytope on n nodes, denoted $\mathcal{P}_A^n$. Clearly $\mathcal{P}_T^n \subseteq \mathcal{P}_S^n \subseteq \mathcal{P}_A^n$. Furthermore, a well-known result states that the dimension of $\mathcal{P}_A^n$ is $n^2 - 3n + 1$ and Grötschel and Padberg [44] show that the dimension of $\mathcal{P}_T^n$ is also $n^2 - 3n + 1$. Therefore, the dimension of $\mathcal{P}_S^n$ is $n^2 - 3n + 1$.

In the next section we probe deeper into the structure of the extreme points of $\mathcal{P}_S^n$.

3.2 Nested Families and Tight Cuts

In this section, we pay special attention to the cut constraints of the ASEP on n nodes. The interaction between these cut constraints leads to some important properties of the extreme points of $\mathcal{P}_S^n$.

Let $x \in \mathcal{P}_S^n$. Any constraint defining the ASEP on n nodes (a node equality (3.1.1) or (3.1.2), a cut constraint (3.1.3), or a nonnegativity constraint (3.1.4)) which holds with equality for x is called a *tight constraint* for x. Let $\emptyset \subset S \subset V$ be such that $2 \le |S| \le n-2$. We say that S is a *tight set* of x if $x(\delta(S)) = 1$. Similarly, if S is a tight set of x then we say that $\delta(S)$ is a *tight cut* of x. We denote by $\Theta(x)$ the set of all the tight cut constraints of x. Given $x \in \mathcal{P}_S^n$ we define $E(x) = \{e \in E \mid x_e > 0\}$ and we then say that the *support digraph* of x, which we denote by $D(x)$, is the digraph with node set V and arc set $E(x)$.

First we start with an elementary property of the points in $\mathcal{P}_S^n$.

Proposition 3.2.1. *Let $x \in \mathcal{P}_S^n$ and let $\emptyset \subset S \subset V$. Then $x(\delta(S)) = x(\delta(V \backslash S))$.*

Proof. By the node equalities (3.1.1) and (3.1.2) we know that

$$\sum_{v \in S} x(\delta(v)) - \sum_{v \in S} x(\delta(V \backslash \{v\})) = |S| - |S|$$

from which the result follows. □

From Theorem 2.3.4 we see that $x \in \mathcal{P}_S^n$ is an extreme point if and only if x is the unique solution to the system of linear equations comprising of all the tight constraints of x. Now, the description of the ASEP on n nodes has precisely $n(n-1)$ variables (one for each arc in the complete digraph on n nodes). By basic linear algebra, we know that such a subsystem must have at least $n(n-1)$ tight constraints. As such, this subsystem must contain at least $n(n-1)$ equations. However, we can remove any redundant equations in our subsystem to get a new subsystem which has exactly $n(n-1)$ equations and which still has x as its unique solution. This is a cobasis for x.

Given a set, A, a *nested family* of subsets of A is a collection $\mathcal{L}$ of subsets of A such that for any two $S_1, S_2 \in \mathcal{L}$ we have that $S_1 \cap S_2 = \emptyset$, $S_1 \subseteq S_2$, or $S_2 \subseteq S_1$. This definition leads us to a very important connection between the extreme points of $\mathcal{P}_S^n$ and their respective tight cut constraints. This connection was first proved by Cornuéjols, Fonlupt, and Naddef [25] in a slightly different context but it also holds for the ASEP. We provide a short proof for completeness.

Theorem 3.2.2. *Let x' be an extreme point of $\mathcal{P}_S^n$. Then x' has a cobasis such that the tight sets of x' corresponding to tight cut constraints in the cobasis form a nested family.*

Proof. Consider building a cobasis for x' in the following way. First, we start with all the tight nonnegativity constraints (constraints (3.1.4) in our description of the ASEP) for x' set to equality. We then add as many of the node equalities (equalities of the form (3.1.1) and (3.1.2)) as possible such that the current set of equations has full row rank. From there, we extend our current set of equations to a cobasis of x' by adding tight cut constraints of x' (constraints of the form (3.1.3)). Let $\mathcal{L}$ denote the set of all the tight sets of x' corresponding to the cut constraints in our cobasis. Assume, without loss of generality, that

$$\sum_{S \in \mathcal{L}} |S|$$

is minimimum among all possible choices for $\mathcal{L}$.

First, we note that for every $S \in \mathcal{L}$ that $|S| \leq \lfloor \frac{n}{2} \rfloor$. Otherwise, by Proposition 3.2.1, since every node equality is either present in the cobasis or implied by the tight nonnegativity constraints and node equalities in the cobasis, we can replace the equation $x(\delta(S)) = 1$ in our cobasis with the equation $x(\delta(V \backslash S)) = 1$ to get a new cobasis for x'. However, this contradicts the minimality of $\sum_{S \in \mathcal{L}} |S|$ since $|S| > \lfloor \frac{n}{2} \rfloor$ and so $|V \backslash S| \leq \lfloor \frac{n}{2} \rfloor$.

Next, suppose for a contradiction that there are $A, B \in \mathcal{L}$ such that $A \cap B$, $A - B$, and $B - A$ are all nonempty and assume, without loss of generality, that $|A| \geq |B|$. Notice that since $|A|, |B| \leq \lfloor \frac{n}{2} \rfloor$ that it immediately follows that $V \backslash (A \cup B)$ is also nonempty.

Now, as formal sums of the variables, we have that

$$\begin{aligned} x(\delta(A)) + x(\delta(B)) &= x(\delta(A-B)) + x(\delta(B-A)) + \sum_{v \in A\cap B} x(\delta(v)) \\ &\quad - \sum_{v\in A\cap B} x(\delta(V\backslash\{v\})) + x(A\cap B; V\backslash(A\cup B)) \\ &\quad + x(V\backslash(A\cup B); A\cap B). \end{aligned}$$

So specifically,

$$\begin{aligned} x'(\delta(A)) + x'(\delta(B)) &= x'(\delta(A-B)) + x'(\delta(B-A)) + \sum_{v \in A\cap B} x'(\delta(v)) \\ &\quad - \sum_{v\in A\cap B} x'(\delta(V\backslash\{v\})) + x'(A\cap B; V\backslash(A\cup B)) \\ &\quad + x'(V\backslash(A\cup B); A\cap B). \end{aligned}$$

However, since $A, B \in \mathcal{L}$, we have that $x'(\delta(A)) = 1$ and $x'(\delta(B)) = 1$. Furthermore, by the node equalities,

$$\sum_{v\in A\cap B} x'(\delta(v)) = |A\cap B|$$

and

$$\sum_{v\in A\cap B} x'(\delta(V\backslash\{v\})) = |A\cap B|.$$

Thus

$$\begin{aligned} 1+1 &= x'(\delta(A-B)) + x'(\delta(B-A)) + |A\cap B| - |A\cap B| \\ &\quad + x'(A\cap B; V\backslash(A\cup B)) + x'(V\backslash(A\cup B); A\cap B) \\ 2 &= x'(\delta(A-B)) + x'(\delta(B-A)) + x'(A\cap B; V\backslash(A\cup B)) \\ &\quad + x'(V\backslash(A\cup B); A\cap B). \end{aligned}$$

But by the cut constraints, we know that $x'(\delta(A-B)) \geq 1$ and $x'(\delta(B-A)) \geq 1$ and by the nonnegativity constraints, we know that $x'(A\cap B; V\backslash(A\cup B)) \geq 0$ and $x'(V\backslash(A\cup B); A\cap B) \geq 0$. Thus it must be that $A-B$ and $B-A$ are tight sets of x' and $x'_e = 0$ for every arc e with its endpoints in $A\cap B$ and $V\backslash(A\cup B)$ respectively. Since all these latter constraints are in our cobasis and every node equality is either present in the cobasis or implied by the tight nonnegativity constraints and node equalities in the cobasis, we can use the fact that

$$\begin{aligned} x(\delta(A)) &= -x(\delta(B)) + x(\delta(A-B)) + x(\delta(B-A)) + \sum_{v \in A\cap B} x(\delta(v)) \\ &\quad - \sum_{v\in A\cap B} x(\delta(V\backslash\{v\})) + x(A\cap B; V\backslash(A\cup B)) \\ &\quad + x(V\backslash(A\cup B); A\cap B). \end{aligned}$$

to remove the constraint $x(\delta(A)) = 1$ from our cobasis and replace it with either $x(\delta(A-B)) = 1$ or $x(\delta(B-A)) = 1$ to get a new cobasis for x'. However, since $A \cap B \neq \emptyset$, we have that $|A-B| < |A|$ and $|B-A| < |B| \leq |A|$ which contradicts the minimality of $\sum_{S\in\mathcal{L}} |S|$.

Thus, for every $A, B \in \mathcal{L}$ it must be that at least one of $A\cap B$, $A-B$, or $B-A$ are empty. Thus A and B are disjoint, $A \subseteq B$, or $B \subseteq A$ and therefore $\mathcal{L}$ is a nested family. □

Vempala and Yannakakis [79] use Theorem 3.2.2 to prove an upper bound on the number arcs in $E(x)$ for any extreme point x of $\mathcal{P}_S^n$. Goemans [40] refines this bound as presented in Theorem 3.2.4. Again, we provide a proof for the sake of completeness using the following well-known result about nested families.

Lemma 3.2.3. *Let V be a set such that $|V| = n$. Any nested family of subsets of V contains at most $2n-1$ subsets.*

Proof. Without loss of generality, let $\mathcal{L}$ be a maximal nested family of subsets of V. We proceed by induction on n. If $n = 1$ then $\mathcal{L}$ can contain at most one subset – V itself. Thus the result holds.

Let $n \geq 2$ and suppose that for all $1 \leq n' \leq n-1$ that any nested family of subsets of a set with n' elements contains at most $2n'-1$ subsets. If $\mathcal{L} - \{V\} = \emptyset$ then $\mathcal{L}$ contains at most 1 subset and, since $n \geq 2$, the result follows. Hence, let S be an inclusionwise maximal subset of $\mathcal{L} - \{V\}$. Note that $|S| < n$. Let

$$\mathcal{L}_1 = \{T \in \mathcal{L} \mid T \subseteq S\}$$

and let

$$\mathcal{L}_2 = \{T \in \mathcal{L} \mid T \cap S = \emptyset\}.$$

Since $\mathcal{L}$ is a nested family of subsets of V, $\mathcal{L}_1$ and $\mathcal{L}_2$ are nested families of subsets of S and $V \backslash S$ respectively. Hence, by our inductive hypothesis, $|\mathcal{L}_1| \leq 2|S| - 1$ and $|\mathcal{L}_2| \leq 2(n - |S|) - 1$.

Furthermore, $\mathcal{L}_1$ and $\mathcal{L}_2$ partition $\mathcal{L} - \{V\}$. Thus,

$$\begin{aligned}
|\mathcal{L} - \{V\}| &= |\mathcal{L}_1| + |\mathcal{L}_2| \\
|\mathcal{L} - \{V\}| &\leq (2|S| - 1) + (2(n - |S|) - 1) \\
|\mathcal{L} - \{V\}| &\leq 2n - 2 \\
|\mathcal{L}| &\leq 2n - 1
\end{aligned}$$

and the result follows. □

Theorem 3.2.4. *If x is an extreme point of $\mathcal{P}_S^n$ then $|E(x)| \leq 3n - 4$.*

Proof. Consider building a cobasis for x starting with as many of the tight non-negativity constraints and node equalities as possible. From there we add tight cut constraints such that the tight sets corresponding to these tight cut constraints form a nested family of subsets as outlined in Theorem 3.2.2.

Now, the set of all the node equalities of the ASEP are linearly dependent which we can easily see since

$$\sum_{v \in V} x(\delta(v)) = \sum_{v \in V} x(\delta(V \backslash \{v\})).$$

Thus our cobasis can contain at most $2n - 1$ node equalities.

Next, our nested family of tight sets contains no singletons. However, we can add any singleton (if they are not already present) to a nested family of subsets and obtain a new family of subsets that is also nested. Hence, as a consequence of Lemma 3.2.3, any nested family of subsets that contains no singletons cannot contain more that $n - 1$ subsets. Likewise, if V itself is not already present in a nested family of subsets, then it can be added to obtain a new family of subsets that is also nested. Similarly, if we take any inclusionwise maximal proper subset, S, of V in a nested family of subsets, then we can add the subset $V\backslash S$, if it is not already present, to obtain a new family of subsets that is also nested. For $n \geq 3$, either S or $V\backslash S$ is not a singleton. Thus, any nested family of subsets that does not contain singletons, does not contain the complement of singletons, does not contain V, and does not contain any subset and its complement, has at most $n-3$ subsets. However, the tight sets corresponding to the tight cut constraints in our cobasis have exactly these properties. Thus, there are at most $n-3$ tight cut constraints in the cobasis.

Since there are at most $2n-1$ node equalities and at most $n-3$ tight cut constraints in our cobasis, the rest of the equalities in our cobasis must be tight nonnegativity constraints. Hence, we see that at most $(2n-1)+(n-3) = 3n-4$ of the arcs of x have positive x-values. Therefore, $|E(x)| \leq 3n-4$. □

In this thesis, we are very interested in exhaustively generating the extreme points of $\mathcal{P}_S^n$. Theorem 3.2.4 is useful since it gives us an upper bound on the number of arcs in the support digraph.

We now turn our attention to new ways of generating extreme points of $\mathcal{P}_S^{n'}$ if we already know some extreme points of $\mathcal{P}_S^n$ where $n' > n$.

3.3 Inserting 1-arcs

In this section, we look at a way of constructing new extreme points from previously-known extreme points. Since this construction is simple, it can be used to drastically reduce the number of extreme points that we need to generate by more time-consuming computations.

Let $x \in \mathcal{P}_S^n$ and let S be a tight set of x. Consider replacing the nodes of S with a single node w and defining a new set of values on the arcs of $K_{n-|S|+1}$ denoted by

$x \downarrow_w (S)$ where

$$(x \downarrow_w (S))_{uv} = \begin{cases} \sum_{s \in S} x_{us} & \text{if } v = w \\ \sum_{s \in S} x_{sv} & \text{if } u = w \\ x_{uv} & \text{otherwise} \end{cases}$$

for each arc uv of $K_{n-|S|+1}$. Essentially, $x \downarrow_w (S)$ is obtained from x by identifying the nodes of S. Figure 3.4 depicts a feasible point of $\mathcal{P}_S^5$ along with a tight set S. Figure 3.5 shows the resulting point of $\mathcal{P}_S^4$ obtained by identifying the nodes of S to a single node w.

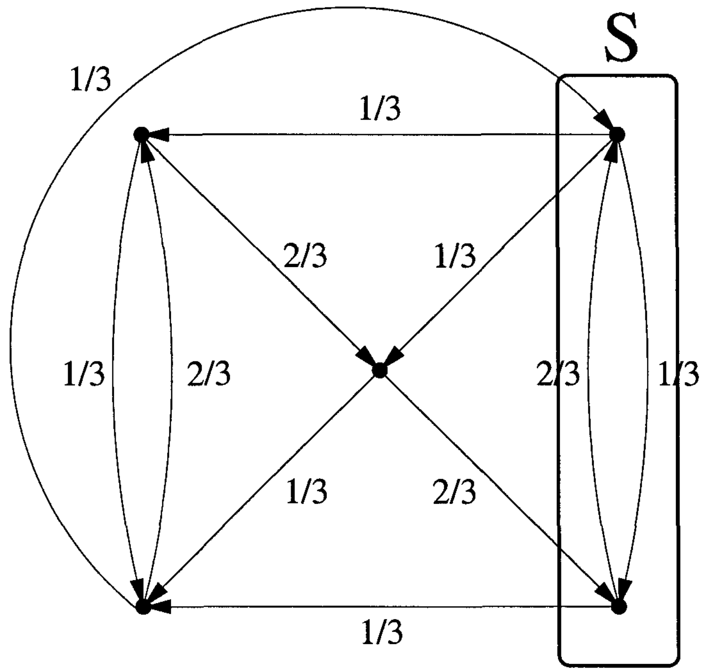

Figure 3.4: A point of $\mathcal{P}_S^5$ with tight set S

Proposition 3.3.1. *Let $x \in \mathcal{P}_S^n$ and let S be a tight set of x. Then $x \downarrow_w (S) \in \mathcal{P}_S^{n-|S|+1}$.*

Proof. Let $K_{n-|S|+1} = (V', E')$ and let $x' = x \downarrow_w (S)$. Since the nonnegativity constraints hold for x they must also hold for x'. Also, for any $v \in V' \backslash \{w\}$ we have that

$$x'(\delta(v)) = x(\delta(v)) = 1$$

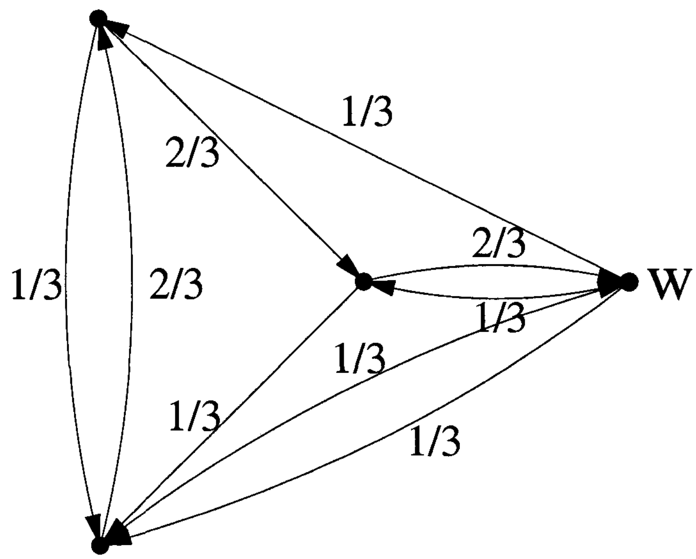

Figure 3.5: The result of contracting S

and

$$x'(\delta(V'\backslash\{v\})) = x(\delta(V\backslash\{v\})) = 1.$$

In addition,

$$x'(\delta(w)) = x(\delta(S)) = 1$$

since S is a tight set of x and

$$x'(\delta(V'\backslash\{w\})) = x(\delta(V\backslash S)) = 1$$

by Proposition 3.2.1. Thus the node equalities hold.

Let $\emptyset \subset T \subset V'$. If $w \in T$ then

$$x'(\delta(T)) = x(\delta((T\backslash\{w\}) \cup S)) \geq 1.$$

If $w \notin T$ then

$$x'(\delta(T)) = x(\delta(T)) \geq 1.$$

In either case, the cut constraints hold.

Therefore, $x' \in \mathcal{P}_S^{n-|S|+1}$. □

Thus we see a way we can create points of the ASEP with smaller number of nodes from those with a larger number of nodes. We now proceed with a way of creating points of the ASEP with a larger number of nodes from those with a smaller number of nodes.

Let $x \in \mathcal{P}_S^n$ and let $w \in V$. Consider replacing the node w with two nodes, u and v. Define $x \uparrow_1^{uv} (w)$ to be the set of arc values applied to the arcs of the complete digraph on $n+1$ nodes as follows:

$$(x \uparrow_1^{uv} (w))_{ab} = \begin{cases} x_{aw} & \text{if } b = u \text{ and } a \neq v \\ x_{wb} & \text{if } a = v \text{ and } b \neq u \\ 0 & \text{if } a = u \text{ and } b \neq v \\ 0 & \text{if } b = v \text{ and } a \neq u \\ 1 & \text{if } ab = uv \\ 0 & \text{if } ab = vu \\ x_{ab} & \text{otherwise.} \end{cases}$$

We call this operation *inserting a 1-arc at w*. Notice that

$$x = (x \uparrow_1^{uv} (w)) \downarrow_w (\{u, v\})$$

and if $x_{uv} = 1$ we have that

$$x = (x \downarrow_w (\{u, v\})) \uparrow_1^{uv} (w)).$$

Figure 3.6 demonstrates the local change that occurs in the support digraph when we insert a 1-arc at w.

Proposition 3.3.2. *Let $x \in \mathcal{P}_S^n$ and let w be a node of $D(x)$. Then $x \uparrow_1^{uv} (w) \in \mathcal{P}_S^{n+1}$.*

Proof. Let $x' = x \uparrow_1^{uv} (w)$ and let $D(x') = (V', E')$. Since the nonnegativity constraints hold for x, they also hold for x'.

For any $a \in V' \backslash \{u, v\}$ we have that $x'(\delta(a)) = x(\delta(a)) = 1$ and $x'(\delta(V' \backslash \{a\})) = x(\delta(V \backslash \{a\})) = 1$. Also $x'(\delta(v)) = x(\delta(w)) = 1$ and $x'(\delta(V' \backslash \{u\})) = x(\delta(V \backslash \{w\})) =$

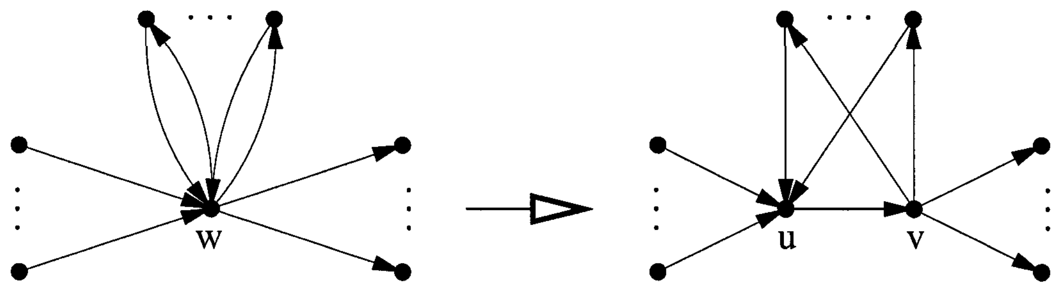

Figure 3.6: Inserting a 1-arc at w

1. Furthermore, $x'(\delta(u)) = x'(\delta(V'\backslash\{v\})) = x'_{uv} = 1$. Thus the node equalities hold for x'.

Let $\emptyset \subset T \subset V'$. If $u, v \notin T$ then $x'(\delta(T)) = x(\delta(T)) \geq 1$. If $u, v \in T$ then $x'(\delta(T)) = x(\delta((T\backslash\{u,v\}) \cup \{w\})) \geq 1$. If $u \in T$ and $v \notin T$ then $uv \in \delta(T)$ and so $x'(\delta(T)) \geq x_{uv} = 1$. Finally, if $u \notin T$ and $v \in T$ then $x'(\delta(T)) = x(\delta((T\backslash\{v\}) \cup \{w\}))$. Hence the cut constraints hold for x' and therefore $x' \in \mathcal{P}_S^{n+1}$. □

Better than just showing that we can create new points in $\mathcal{P}_S^n$ by inserting a 1-arc, this operation also preserves extreme points of $\mathcal{P}_S^n$. What this means is that if we have all the extreme points of $\mathcal{P}_S^n$ then we can immediately construct all of the extreme points of $\mathcal{P}_S^{n+1}$ which have a 1-arc. Thus, we only have to worry about generating the extreme points of $\mathcal{P}_S^{n+1}$ which have no 1-arcs.

Theorem 3.3.3. *Let $x \in \mathcal{P}_S^n$ and let w be a node of $D(x)$. Then $x \uparrow_1^{uv} (w)$ is an extreme point of $\mathcal{P}_S^{n+1}$ if and only if x is an extreme point of $\mathcal{P}_S^n$.*

Proof. Let $x' = x \uparrow_1^{uv} (w)$ and let $D(x') = (V', E')$.

Suppose x' is an extreme point of $\mathcal{P}_S^{n+1}$ and suppose that $x = \frac{1}{2}y + \frac{1}{2}z$ where $y, z \in \mathcal{P}_S^n$. Let $y' = y \uparrow_1^{uv} (w)$ and $z' = z \uparrow_1^{uv} (w)$. Then $x' = \frac{1}{2}y' + \frac{1}{2}z'$ and since x' is an extreme point we have that $y' = z' = x'$. Hence, $y = z = x$ and therefore, x is an extreme point of $\mathcal{P}_S^n$.

Now suppose that x is an extreme point of $\mathcal{P}_S^n$ and that $x' = \frac{1}{2}y' + \frac{1}{2}z'$ where

$y', z' \in \mathcal{P}_S^{n+1}$. Notice that

$$\begin{aligned} x'(\delta(\{u,v\})) &= x'(\delta(u)) - x'_{uv} + x'(\delta(v)) - x'_{vu} \\ &= 1 - 1 + 1 - 0 \\ &= 1 \end{aligned}$$

and so $\{u, v\}$ is a tight set of x' (and hence of y' and z' too). Thus by Proposition 3.3.1 we can let $y = y' \downarrow_w (\{u, v\})$ and let $z = z' \downarrow_w (\{u, v\})$. Then we have that $x = \frac{1}{2}y + \frac{1}{2}z$ so $y = z = x$ since x is an extreme point. Thus, y' and z' can differ from x' only in their values on arcs incident to u or v. Now for any arc ab such that $x'_{ab} = 0$ we must have that $y'_{ab} = z'_{ab} = 0$ since $x' = \frac{1}{2}y' + \frac{1}{2}z'$. Now $x'_{ua} = 0$ and $x'_{av} = 0$ for every $a \in V' \backslash \{u, v\}$. Hence we can further surmise that y' and z' can differ from x' only in their values on the arcs uv and vu. However, since $x'_{vu} = 0$ we know that $y'_{vu} = z'_{vu} = 0$. As well, since $x'_{uv} = 1$ and $x' = \frac{1}{2}y' + \frac{1}{2}z'$ so $y'_{uv} = z'_{uv} = 1$. Thus $y' = z' = x'$ and therefore x' is an extreme point of $\mathcal{P}_S^{n+1}$. □

Notice that Theorem 3.3.3 can be restated in the following way.

Corollary 3.3.4. *Let $x \in \mathcal{P}_S^n$ where $n \geq 4$ and let $uv \in E(x)$ such that $x_{uv} = 1$. Then x is an extreme point of $\mathcal{P}_S^n$ if and only if $x \downarrow_w (\{u, v\})$ is an extreme point of $\mathcal{P}_S^{n-1}$.*

This leads us to the following corollary.

Corollary 3.3.5. *Let x be an extreme point of $\mathcal{P}_S^n$ with exactly k 1-arcs. If $0 < k < n$ then there at least $2(n-k)$ arcs of $E(x)$ which are not integer and there are at least four arcs of $E(x)$ which have x-values in the interval $[\frac{1}{2}, 1)$.*

Proof. Let x' be the extreme point obtained from x by sequentially contracting all the 1-arcs. Due to the node equalities, for any arc $uv \in E(x)$ where $x_{uv} = 1$ we have that $x_{ua} = 0$ and $x_{av} = 0$ for any $a \in V \backslash \{u, v\}$. Thus contracting any 1-arc of x leaves all the non-integer arc-values unchanged. By repeatedly applying this argument, we see that there is a bijection between the arcs of $E(x)$ with non-integer values and the arcs of $E(x')$. By Theorem 3.3.3, x' is an extreme point of $\mathcal{P}_S^{n-k}$. Since x' has no 1-arcs we know that every node of $D(x')$ has out-degree at least 2. Thus $|E(x')| \geq 2(n-k)$.

Secondly, from Theorem 3.2.4 we know that $|E(x')| \leq 3(n-k)-4$. Suppose, for a contradiction, that $D(x')$ has $t \leq 3$ nodes of out-degree exactly 2. Hence, all the other nodes of $D(x')$ have out-degree at least 3. By adding up the out-degrees of all nodes we see that

$$\begin{aligned} |E(x')| &\geq 2t + 3((n-k)-t) \\ &= 3(n-k)-t \\ &\geq 3(n-k)-3 \end{aligned}$$

which is a contradiction. Thus, at least four of the nodes of $D(x')$ must have out-degree exactly 2. For any such node, either both the arcs which have their tails at the node have x'-values of $\frac{1}{2}$ or one of the arcs has an x'-value in the interval $(\frac{1}{2}, 1)$. □

In the next section, we consider another operation which can create new extreme points of $\mathcal{P}_S^{n+1}$ from a certain class of extreme points of $\mathcal{P}_S^n$.

3.4 Half-integer Points

We say that $x \in \mathcal{P}_S^n$ is a *half-integer* point if $x_{uv} \in \{0, \frac{1}{2}, 1\}$ for every arc uv of the complete digraph K_n.

Let x be a half-integer point of $\mathcal{P}_S^n$ and let w be a node of $D(x)$. Consider the multi-digraph obtained from $D(x)$ by duplicating all the 1-arcs of $E(x)$. Let aw, wb, cw, and wd be the four arcs incident to w in this multi-digraph. Consider replacing w with two nodes u and v and creating some new values on the arcs of the resulting complete digraph on $n+1$ nodes, denoted $x \uparrow_2^{uv} (aw, wb, cw, wd)$, and defined as follows.

$$(x \uparrow_2^{uv} (aw, wb, cw, wd))_{pq} = \begin{cases} x_{pq} & \text{if } p, q \in V \backslash \{u, v\} \\ \frac{1}{2} & \text{if } pq \in \{au, ub, cv, vd, uv, vu\} \\ 0 & \text{otherwise} \end{cases}$$

for each arc pq of K_{n+1}. We call this operation a *2-jack*. The local changes which occur as a result of a 2-jack are demonstrated in Figure 3.7. Here we chose a node w which is incident to a 1-arc to show how the operation splits this arc.

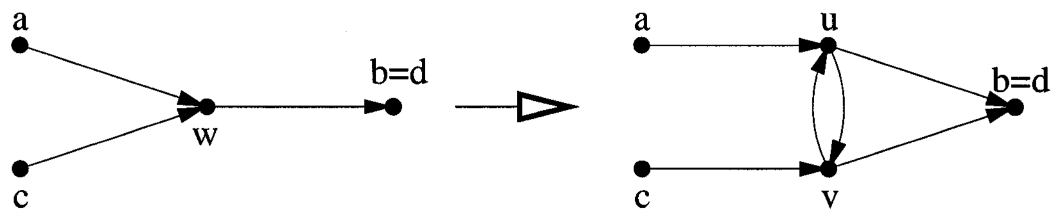

Figure 3.7: A 2-jack applied at w

Proposition 3.4.1. *Let x be a half-integer point of $\mathcal{P}_S^n$ and let w be a node of $D(x)$. Then $x \uparrow_2^{uv} (aw, wb, cw, wd)$ is a half-integer point of $\mathcal{P}_S^{n+1}$.*

Proof. Let $x' = x \uparrow_2^{uv} (aw, wb, cw, wd)$ and let $D(x') = (V', E')$. Clearly the nonnegativity constraints hold for x' since they do for x and x' is a half-integer point.

If $p \in V' \backslash \{u, v\}$ then $x'(\delta(p)) = x(\delta(p)) = 1$ and $x'(\delta(V' \backslash \{p\})) = x(\delta(V \backslash \{p\})) = 1$. By construction, $x'(\delta(u)) = 1$, $x'(\delta(v)) = 1$, $x'(\delta(V' \backslash \{u\})) = 1$, and $x'(\delta(V' \backslash \{v\})) = 1$, so the node equalities hold.

Let $\emptyset \subset T \subset V'$ where $2 \leq |T| \leq n-2$. If $u, v \notin T$ then $x'(\delta(T)) = x(\delta(T)) \geq 1$. If $u, v \in T$ then $x'(\delta(T)) = x(\delta((T \backslash \{u, v\}) \cup \{w\})) \geq 1$. If $u \in T$ and $v \notin T$ then

$$\begin{aligned} x'(\delta(T)) &\geq x'(\delta(T \backslash \{u\})) - x'_{au} + x'_{uv} \\ &= x'(\delta(T \backslash \{u\})) \\ &= x(\delta(T \backslash \{u\})) \\ &\geq 1. \end{aligned}$$

Similarly, if $v \in T$ and $u \notin T$ then

$$\begin{aligned} x'(\delta(T)) &\geq x'(\delta(T\backslash\{v\})) - x'_{cv} + x'_{vu} \\ &= x'(\delta(T\backslash\{v\})) \\ &= x(\delta(T\backslash\{v\})) \\ &\geq 1. \end{aligned}$$

Hence, the cut constraints hold and therefore $x' \in \mathcal{P}_S^{n+1}$. □

Better yet, we can use this operation to generate more half-integer extreme points as follows.

Theorem 3.4.2. *Let x be a half-integer extreme point of $\mathcal{P}_S^n$ and let w be a node of $D(x)$. Then $x \uparrow_2^{uv} (aw, wb, cw, wd)$ is an extreme point of $\mathcal{P}_S^{n+1}$ if and only if w is not incident to two 1-arcs of $E(x)$.*

Proof. Let $x' = x \uparrow_2^{uv} (aw, wb, cw, wd)$ and let $D(x')$ have node set V'.

If w is incident to two 1-arcs of $E(x)$ then these are the only two arcs of $E(x)$ which are incident to w. Hence, $a = c$ and $b = d$. Define two sets of arc-values, y' and z' respectively on the arcs of K_{n+1} as follows.

$$y'_{pq} = \begin{cases} x_{pq} & \text{if } p, q \in V\backslash\{u, v\} \\ 1 & \text{if } pq \in \{au, uv, vb\} \\ 0 & \text{otherwise} \end{cases}$$

$$z'_{pq} = \begin{cases} x_{pq} & \text{if } p, q \in V\backslash\{u, v\} \\ 1 & \text{if } pq \in \{av, vu, ub\} \\ 0 & \text{otherwise.} \end{cases}$$

Then $x' = \frac{1}{2}y' + \frac{1}{2}z'$. However, $y' = x \uparrow_1^{uv} (w)$ and $z' = x \uparrow_1^{vu} (w)$ and so $y', z' \in \mathcal{P}_S^{n+1}$ by Theorem 3.3.2. Additionally, $z' \neq y'$ so x' is not an extreme point of $\mathcal{P}_S^{n+1}$.

Now suppose that x' is not an extreme point of $\mathcal{P}_S^{n+1}$. Hence let $x' = \frac{1}{2}y' + \frac{1}{2}z'$ where $y', z' \in \mathcal{P}_S^{n+1}$ and $z' \neq y'$. Now let $\emptyset \subset S \subset V'$ be such that $x'(\delta(S)) = 1$. But $y', z' \in \mathcal{P}_S^{n+1}$ so $y'(\delta(S)) \geq 1$ and $z'(\delta(S)) \geq 1$ so $\frac{1}{2}y'(\delta(S)) + \frac{1}{2}z'(\delta(S)) \geq 1$. However, $x'(\delta(S)) = \frac{1}{2}y'(\delta(S)) + \frac{1}{2}z'(\delta(S))$ so S must be a tight set in y' and z'. Specifically, $\{u, v\}$ is a tight set of x' so it is also a tight set of y' and z'.

Let $y = y' \downarrow_w (\{u, v\})$ and $z = z' \downarrow_w (\{u, v\})$. Then $x = \frac{1}{2}y + \frac{1}{2}z$ and so $y = z = x$ since x is an extreme point. Hence, $y'_{pq} = z'_{pq} = x'_{pq}$ for any arc of $D(x')$ which is not incident to either u or v.

We now focus our attention on y'. Notice that

$$\begin{aligned} y'_{uv} + y'_{vu} &= y'(\delta(u)) + y'(\delta(v)) - y'(\delta(\{u, v\})) \\ &= 1 + 1 - 1 \\ &= 1. \end{aligned}$$

Hence, if $y'_{uv} = \alpha$ then $y'_{vu} = 1 - \alpha$. As noted in the proof of Theorem 3.3.3, if $pq \in E(y')$ then $pq \in E(x')$. Thus the only possible arcs of $E(y')$ which are incident to either u or v are au, ub, cv, vd, uv, and vu. So, by the node equalities for u and v we have that

$$\begin{aligned} y'_{uv} &= \alpha, \\ y'_{vu} &= 1 - \alpha, \\ y'_{au} &= \alpha, \\ y'_{ub} &= 1 - \alpha, \\ y'_{cv} &= 1 - \alpha, \text{ and} \\ y'_{vd} &= \alpha. \end{aligned}$$

If $a \neq c$ then w is the head of precisely two arcs of $E(x)$. Hence, $x_{aw} = \frac{1}{2}$ and $x_{cw} = \frac{1}{2}$ since x is a half-integer point. Thus

$$\sum_{p \in V \setminus \{a, w\}} x_{ap} = \frac{1}{2}$$

and so

$$\sum_{p \in V \setminus \{a,u,v\}} x'_{ap} = \frac{1}{2},$$

which implies that

$$\sum_{p \in V \setminus \{a,u,v\}} y'_{ap} = \frac{1}{2}.$$

Therefore, $y'_{au} = \frac{1}{2}$ by the node equalities. This implies that $\alpha = \frac{1}{2}$ and hence $y' = x'$. By symmetry, if $b \neq d$ then $y' = x'$. However, all these arguments apply to z' too so if $a \neq c$ or $b \neq d$ then $z' = x'$ and hence $y' = z'$ which is a contradiction. We can conclude therefore that $a = c$ and $b = d$ and so w is incident to exactly two 1-arcs in $D(x)$. □

Unfortunately, inserting a 1-arc and the 2-jack operations do not generate all of the half-integer extreme points of the ASEP. Figure 3.8 depicts the support digraph of a half-integer extreme point of $\mathcal{P}_S^{10}$ which has no 1-arc and no pair of anti-parallel arcs. Hence this extreme point cannot be obtained by applying the operations outlined in this chapter to some extreme point of $\mathcal{P}_S^9$.

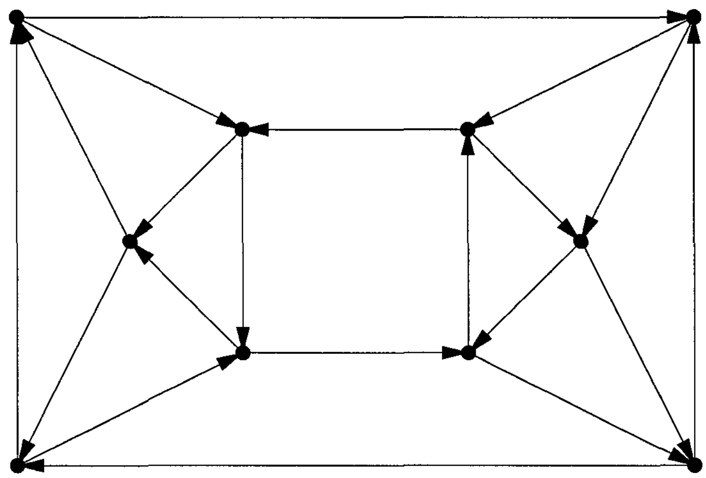

Figure 3.8: A half-integer extreme point not generated by the operations

In the next chapter, we use these results about extreme points of the ASEP to consider a measure of the gap in between the ASEP on n nodes and $\mathcal{P}_T^n$.

Chapter 4

The Integrality Gap of the Subtour Elimination Formulation

In this chapter, we discuss a measure called the integrality gap that is the focus of the remainder of this thesis. In essence, we are trying to quantify how good a lower bound we can find for the optimal value of the ATSP by optimizing over the ASEP. We see in the first section that optimizing over the ASEP does not provide a good bound at all when we are considering arbitrary nonnegative cost functions. However, if we restrict our attention to a special class of cost functions then we can obtain more reasonable results. We describe in this chapter how we compute the integrality gap for small values of n. We finish the chapter by showing the relationship between the 1-arc insertion operation and the integrality gap.

4.1 Triangle Inequalities

Although the ATSP can be defined for any set of nonnegative arc costs, many researchers have limited their attention to nonnegative arc costs which obey the *triangle inequalities* which are defined as follows:

$$c_{uv} \leq c_{uw} + c_{wv} \text{ for all } uv \in E,\ w \in V \backslash \{u, v\}$$

where $K_n = (V, E)$ is the complete digraph on n nodes. We also say that any set of nonnegative arc costs on the arcs of the complete digraph which obey the triangle inequalities is *metric* and we call any instance of the ATSP with metric costs the *metric ATSP*.

In 1982, Frieze, Galbiati, and Maffioli [35] invented a polynomial time algorithm which could find a solution to the metric ATSP whose total cost was within a factor of $\lceil \log_2(n) \rceil$ of the cost of an optimal solution. Two decades later, Bläser [10] found a similar algorithm but with a factor of roughly $0.999 \log_2(n)$. Most recently, Kaplan, Lewenstein, Shafrir, and Sviridenko [52] improved the algorithm to obtain a factor of roughly $0.842 \log_2(n)$. Ultimately, we would like to find a polynomial time algorithm which could find a solution to the metric ATSP whose cost would be within a constant factor k (which does not depend on n) of the cost of an optimal solution. Any such algorithm is called a *k-approximation algorithm* for the metric ATSP. Papadimitriou and Vempala [73] show that if a k-approximation algorithm does exist for the metric ATSP then $k > \frac{117}{116}$ unless P = NP.

It is worth noting that a $\frac{3}{2}$-approximation algorithm does exist for the metric STSP. It was presented by Christofides [22] in 1976. Astonishingly enough, no k-approximation algorithm has been developed for the metric STSP in the last thirty years with $k < \frac{3}{2}$.

Returning now to the ATSP, consider a strengthening the triangle inequalities so that we have

$$c_{uv} \leq \alpha(c_{uw} + c_{wv})$$

for some $\alpha \in [\frac{1}{2}, 1)$ and every distinct triple u, v, w of nodes of K_n. With these strengthened triangle inequalities, Chandran and Ram [17] give a $\frac{\alpha}{1-\alpha}$-approximation for the resulting ATSP.

There is a way to create a set of metric costs on the complete digraph on n nodes from certain costs assigned to the arcs of a strongly connected spanning subgraph. This process is described by the following well-known lemma.

Lemma 4.1.1. *Let $K_n = (V, E)$ be the complete digraph on n nodes and let $G = (V, E')$ be a strongly connected spanning subgraph of K_n. Let c be a set of nonnegative arc costs assigned to the arcs of G such that for every $uv \in E'$, c_{uv} is at most the cost of a minimum cost (u, v)-dipath in G. Define a set of costs c' on the arcs of K_n where $c'_{uv} = c_{uv}$ if $uv \in E'$ and c'_{uv} is the cost of a minimum cost (u, v)-dipath in G if $uv \in E \backslash E'$. Then c' is metric.*

Proof. Since G is spanning and strongly connected, c' is well-defined. Let $uv \in E$ and let $w \in V\backslash\{u, v\}$. Let P_{uw} be a minimum cost (u, w)-dipath in G and let P_{wv} be a minimum cost (w, v)-dipath in G (either of these dipaths could be a single arc). Now, the union of P_{uw} and P_{uv} is a (u, v)-walk in G. Hence there is a (u, v)-dipath in G of cost at most $c(P_{uw}) + c(P_{wv})$. Thus

$$c'_{uv} \leq c(P_{uw}) + c(P_{wv}).$$

However, $c'_{uw} = c(P_{uw})$ and $c'_{wv} = c(P_{wv})$. Therefore,

$$c'_{uv} \leq c'_{uw} + c'_{wv}$$

and so c' is metric. □

We call c', as described in Lemma 4.1.1, the *metric completion* of c. We see in the next section why it makes sense to limit our attention to metric cost functions when comparing the ATSP to optimizing over the ASEP.

4.2 The Integrality Gap

In this section, we describe a way of measuring the difference between the ATSP and optimizing over the ASEP. Since the solutions of the ATSP are always integer and those found by optimizing over the ASEP are not necessarily integer, this measure is known as the *integrality gap*. We define this measure more formally later in this section once we have established the necessary notation.

Let c be a nonnegative set of costs assigned to the arcs of K_n. We define the linear program denoted ATSP_{LP} by

$$\begin{array}{llcll} \text{minimize} & cx & & & \\ \text{subject to:} & x(\delta(v)) & = & 1 & \text{for all } v \in V \\ & x(\delta(V\backslash\{v\}) & = & 1 & \text{for all } v \in V \\ & x(\delta(S)) & \geq & 1 & \text{for all } \emptyset \subset S \subset V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

Notice that ATSP_{LP} is just the linear programming relaxation of our formulation of the ATSP and is also just optimizing over the ASEP.

Let ATSP(c) denote the optimal value of the ATSP with respect to the cost function c and let $\text{ATSP}_{LP}(c)$ denote the optimal value of the ATSP_{LP} with respect to c. Then $\text{ATSP}_{LP}(c) \leq \text{ATSP}(c)$. Held and Karp [48, 49] used this fact in their 1970 and 1971 papers to create a branch and bound procedure for solving the STSP, but they added as a footnote that the same ideas could be applied to the asymmetric case.

For a nonnegative cost function c we define

$$\text{Gap}(c) = \frac{\text{ATSP}(c)}{\text{ATSP}_{LP}(c)}$$

to be the *integrality gap* of c. We see in Proposition 4.2.1 a known result that for each $n \geq 5$ and any $M > 0$ there exists a nonnegative cost vector c on K_n such that $\text{ATSP}_{LP}(c) = n$ but $\text{ATSP}(c) \geq M$. Thus, for $n \geq 5$ there is no upper bound on the values of the integrality gaps of nonnegative cost vectors defined on the arcs of K_n. However, it is important to note that these arc costs are not metric.

Proposition 4.2.1. *For every $n \geq 5$ and any $M > 0$ there is a nonnegative cost function c on K_n such that $ATSP_{LP}(c) = n$ but $ATSP(c) \geq M$.*

Proof. Consider the extreme point $x \in \mathcal{P}_S^5$ depicted in Figure 4.1 where every arc shown has an x-value of $\frac{1}{2}$ and those not shown have an x-value of 0. The support digraph of x is not hamiltonian. We assign a cost of 1 to each arc in $E(x)$ and a cost of M to all other arcs. Then

$$\begin{aligned} cx &= \sum_{uv \in E(x)} x_{uv} \\ &= \sum_{u \in V} x(\delta(u)) \\ &= 5. \end{aligned}$$

However, since $D(x)$ is not hamiltonian, every tour of K_5 must contain at least one arc not in $E(x)$. Hence $\text{ATSP}(c) \geq M$ and the result follows for $n = 5$.

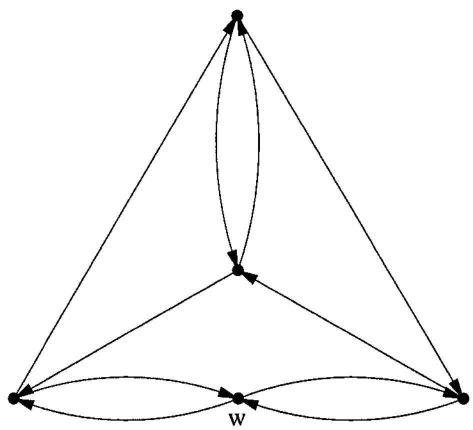

Figure 4.1: A non-hamiltonian digraph on 5 nodes

Now consider repeatedly inserting a 1-arc at w as depicted in Figure 4.2 where each thick arc has an x-value of 1 and all other arcs shown have x-values of $\frac{2}{2}$. By Theorem 3.3.3 the resulting points are extreme points of the ASEP. Furthermore, each of these extreme points is also non-hamiltonian. If we again assign a cost of 1 to every arc in the support digraph and M to those not in the support digraph then, again, the result follows for each $n \geq 5$.

□

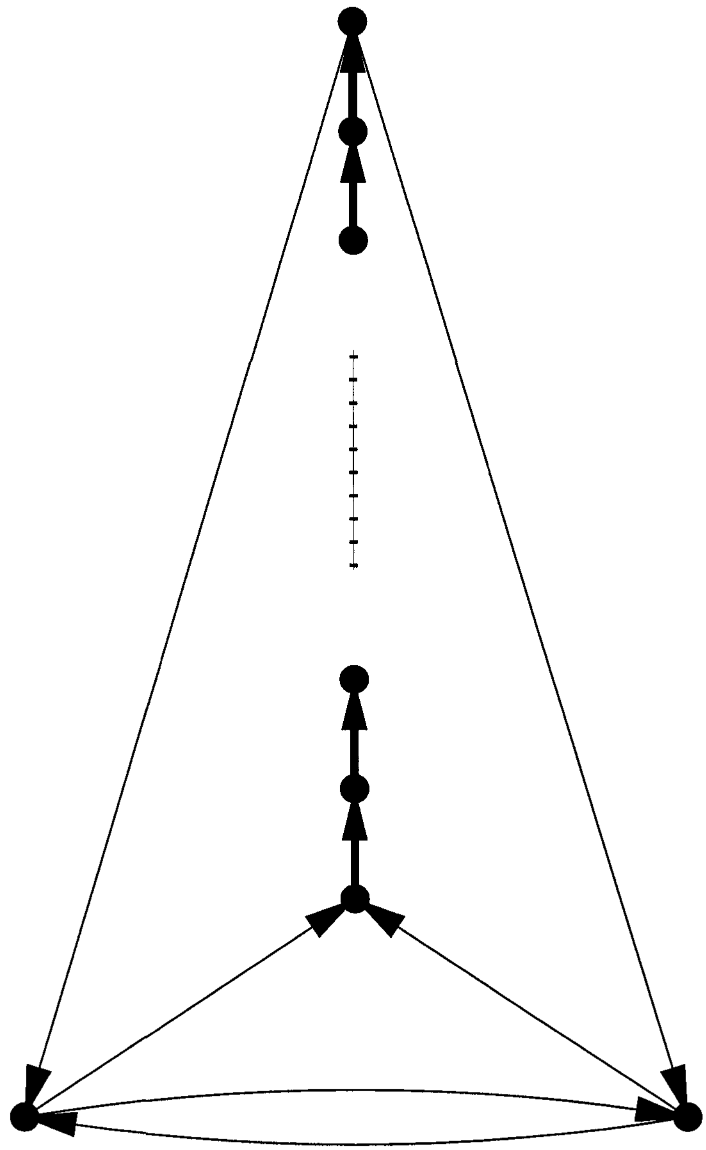

Figure 4.2: A non-hamiltonian digraph

Hence we define the *Integrality Gap* on $K_n = (V, E)$ to be

$$\text{Gap}_n = \max\{\text{Gap}(c) \mid c \in \mathbb{R}^E,\ c \text{ metric},\ c \neq 0\}.$$

We now show that Gap_n is well-defined for each value of n.

Proposition 4.2.2. *Let c be a metric cost vector defined on the arcs of the complete digraph on n nodes. Then $ATSP_{LP}(c) = 0$ if and only if $c = 0$.*

Proof. Let $K_n = (V, E)$ and let x be an optimal solution which has a cost of $\text{ATSP}_{LP}(c) = 0$. Then $c_e = 0$ whenever $e \in E(x)$. Since x obeys the cut constraints, it must be that $D(x)$ is strongly connected. Since c is metric, for any $uv \in E \backslash E(x)$, c_{uv} is at most the cost of a minimum cost (u, v)-dipath in $D(x)$. However, any (u, v)-dipath in $D(x)$ has cost 0 so $c_{uv} = 0$. Therefore $c = 0$. □

We are interested in finding some bounds on Gap_n. Williamson [81] proves that for every n that $\text{Gap}_n \leq \lceil \log_2(n) \rceil$. This bound is unsatisfactory for many researchers since for the symmetric case, the corresponding integrality gap is at most $\frac{3}{2}$ [82, 77]. It was conjectured [16] that $\text{Gap}_n \leq \frac{4}{3}$ for every $n \geq 3$. We refute this conjecture in the next chapter by presenting a family of metric costs whose integrality gaps can be made arbitrarily close to 3/2. Independently, Charikar, Goemans, and Karloff [18] show that for any $\epsilon > 0$ there is a (usually large) value of n such that there exists a set of arc costs, c, on K_n where $\text{Gap}(c) > 2 - \epsilon$. In this chapter and the next, we compute a lower bound on Gap_n for $4 \leq n \leq 15$ and we use patterns we see in these results along with the recursion idea of Charikar, Goemans, and Karloff to create our own family of arc costs whose integrality gaps approach 2. Both our result and that of Charikar, Goemans, and Karloff refutes the earlier conjecture and shows that

$$\max_{n \geq 3} \{\text{Gap}_n\} \geq 2.$$

We begin our search by computing Gap_n for small values of n. Since computing Gap_n from its definition would require us to check an infinite number of nonzero metric cost vectors, we first need to find a new method to calculate this quantity. We use the same framework as that used in Boyd and Labonté [14] for the STSP or by Cheung, Cunningham, and Tang [19], in which the objective value of the ATSP is normalized.

Lemma 4.2.3. *For any metric costs c assigned to the arcs of the complete digraph on n nodes such that $c \neq 0$, there are metric costs c' assigned to the arcs of the complete digraph on n nodes such that $ATSP(c') = 1$ and $Gap(c') = Gap(c)$.*

Proof. By Proposition 4.2.2, since c is metric and $c \neq 0$ it must be that $\text{ATSP}(c) > 0$. Define $c' = \frac{1}{\text{ATSP}(c)}c$. It is easy to show that c' is non-zero and metric. Furthermore, $\text{ATSP}(c') = 1$ and $\text{ATSP}_{LP}(c') = \frac{\text{ATSP}_{LP}(c)}{\text{ATSP}(c)}$. Thus

$$\begin{aligned} \text{Gap}(c') &= \frac{1}{\frac{\text{ATSP}_{LP}(c)}{\text{ATSP}(c)}} \\ &= \frac{\text{ATSP}(c)}{\text{ATSP}_{LP}(c)} \\ &= \text{Gap}(c). \end{aligned}$$

□

Hence we see that

$$\text{Gap}_n = \max\{\frac{1}{\text{ATSP}_{LP}(c)} \mid c \in \mathbb{R}^E,\ c \text{ metric},\ \text{ATSP}(c) = 1\}.$$

However, if $c(T) > 1$ for every tour T of K_n then, by Lemma 4.2.3, there is a non-zero metric set of arc costs, c', such that $\text{ATSP}(c') = 1$ and $\text{ATSP}_{LP}(c') < \text{ATSP}_{LP}(c)$. Thus

$$\begin{aligned} \text{Gap}_n &= \max\{\frac{1}{\text{ATSP}_{LP}(c)} \mid c \in \mathbb{R}^E,\ c \text{ metric},\ c(T) \geq 1 \text{ for all tours } T \text{ of } K_n\}, \\ \frac{1}{\text{Gap}_n} &= \min\{\text{ATSP}_{LP}(c) \mid c \in \mathbb{R}^E,\ c \text{ metric},\ c(T) \geq 1 \text{ for all tours } T \text{ of } K_n\}. \end{aligned}$$

Now, we know from Proposition 2.3.3 that for any set of nonnegative arc costs c, $\text{ATSP}_{LP}(c)$ is attained by an extreme point of $\mathcal{P}_S^n$. Let $\mathcal{X}_S^n$ denote the set of extreme points of $\mathcal{P}_S^n$. Hence we can state that for any nonnegative set of arc costs

$$\text{ATSP}_{LP}(c) = \min_{x \in \mathcal{X}_S^n}\{cx\}.$$

Therefore,

$$\begin{aligned} \frac{1}{\text{Gap}_n} &= \min\{\min_{x \in \mathcal{X}_S^n}\{cx\} \mid c \text{ metric},\ c(T) \geq 1 \text{ for all tours } T\} \\ &= \min_{x \in \mathcal{X}_S^n}\{\min\{cx \mid c \text{ metric},\ c(T) \geq 1 \text{ for all tours } T,\ x \text{ is optimal w.r.t. } c\}\}. \end{aligned}$$

Notice that the constraints that "c must be metric" and that "$c(T) \geq 1$ for each tour T of K_n" are linear constraints. Hence, if we can somehow find a set of linear constraints that are equivalent to "x is optimal with respect to c" then for a fixed x,

$$\min\{cx \mid c \text{ metric}, \; c(T) \geq 1 \text{ for all tours } T, \; x \text{ is optimal w.r.t. } c\} \tag{4.2.1}$$

is a linear program. In order to find such a set of linear constraints, we need to consider the *dual* of the ATSP$_{LP}$.

Let $K_n = (V, E)$ and let $\mathcal{S} = \{S \subset V \mid 2 \leq |S| \leq n-2\}$. Create a nonnegative variable d_S for each $S \in \mathcal{S}$ and also create two variables, y_v^{out} and y_v^{in}, for each $v \in V$. Then the dual of the ATSP$_{LP}$ is

$$\begin{array}{llll} \text{maximize} & \sum_{v \in V} y_v^{\text{out}} + \sum_{v \in V} y_v^{\text{in}} + \sum_{S \in \mathcal{S}} d_S & & \\ & & & \\ \text{subject to:} & y_u^{\text{out}} + y_v^{\text{in}} + \sum_{S \in \mathcal{S}} (d_S \mid uv \in \delta(S)) & \leq \; c_{uv} & \forall \; uv \in E \\ & & & \\ & d_S & \geq \; 0 & \forall \; S \in \mathcal{S}. \end{array}$$

Now, by duality theory we get the following result directly from Theorem 2.4.3 applied to the ATSP$_{LP}$.

Theorem 4.2.4. *Let x be a feasible solution to the ATSP$_{LP}$. Then x is optimal with respect to c if and only if there is a feasible solution, (y, d) to the above dual such that whenever $x_{uv} > 0$ we have that*

$$y_u^{out} + y_v^{in} + \sum_{S \in \mathcal{S}} (d_S \mid uv \in \delta(S)) = c_{uv}$$

and whenever $x(\delta(S)) > 1$ we have that

$$d_S = 0.$$

For a point $x \in \mathcal{P}_S^n$ let $\mathcal{S}(x)$ denote the set of tight sets of x. Thus x is an optimal solution of the ATSP_{LP} if and only if there is a (y, d) such that

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{S \in \mathcal{S}(x)} (d_S \mid uv \in \delta(S)) \leq c_{uv} \quad \forall\, uv \in E \backslash E(x) \tag{4.2.2}$$

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{S \in \mathcal{S}(x)} (d_S \mid uv \in \delta(S)) = c_{uv} \quad \forall\, uv \in E(x) \tag{4.2.3}$$

$$d_S \geq 0 \quad \forall\, S \in \mathcal{S}(x) \tag{4.2.4}$$

Hence we can replace the condition that "x is optimal with respect to c" in (4.2.1) with the linear constraints (4.2.2), (4.2.3), and (4.2.4). Thus, given $x \in \mathcal{X}_S^n$ let $\text{Gap}(x)$ denote the optimal value of the following linear program:

$$\begin{array}{lrcll}
\text{minimize} & \sum_{uv \in E} x_{uv} c_{uv} & & & \\
\text{subject to:} & c_{uw} + c_{wv} - c_{uv} & \geq & 0 & \forall\, uv \in E,\ w \in V \backslash \{u, v\} \\
 & c(T) & \geq & 1 & \forall \text{ tours } T \text{ of } K_n \\
\multicolumn{2}{r}{c_{uv} - y_u^{\text{out}} - y_v^{\text{in}} - \sum_{S \in \mathcal{S}(x)} (d_S \mid uv \in \delta(S))} & \geq & 0 & \forall\, uv \in E \backslash E(x) \\
\multicolumn{2}{r}{c_{uv} - y_u^{\text{out}} - y_v^{\text{in}} - \sum_{S \in \mathcal{S}(x)} (d_S \mid uv \in \delta(S))} & = & 0 & \forall\, uv \in E(x) \\
 & c_{uv} & \geq & 0 & \forall\, uv \in E \\
 & d_S & \geq & 0 & \forall\, S \in \mathcal{S}(x).
\end{array}$$

Thus we have that

$$\frac{1}{\text{Gap}(x)} = \min\{cx \mid c \text{ metric},\ c(T) \geq 1 \text{ for all tours } T,\ x \text{ is optimal w.r.t. } c\}$$

and therefore,

$$\frac{1}{\text{Gap}_n} = \min_{x \in \mathcal{X}_S^n} \{\text{Gap}(x)\}.$$

This gives us an algorithm for computing Gap_n as outlined in Algorithm 4.2.5 below.

Algorithm 4.2.5.

Input: An integer $n \geq 3$

Output: Gap_n

Generate all the extreme points in $\mathcal{X}_S^n$
for each $x \in \mathcal{X}_S^n$
    Compute $\text{Gap}(x)$
Compute $\text{Gap}_n = 1/(\min_{x \in \mathcal{X}_S^n}\{\text{Gap}(x)\})$

Although Algorithm 4.2.5 does exactly compute Gap_n, there are still a few difficulties. Firstly, even for small values of n, there is a huge number of points in $\mathcal{X}_S^n$ and generating them is a very difficult task. Secondly, even though computing $\text{Gap}(x)$ amounts to solving a linear program for a given extreme point x, these linear programs have a large number of constraints. There are $(n-1)!$ tour constraints (constraints of the form $c(T) \geq 1$ for each tour T of K_n) in the linear program with no clear separation oracle. We explain in future sections how we attempt to deal with these difficulties.

4.3 Generating $\mathcal{X}_S^n$ for $3 \leq n \leq 6$

For each $3 \leq n \leq 6$, the method for generating the extreme points of $\mathcal{X}_S^n$ and subsequently computing the integrality gap was the same. To generate the extreme points of $\mathcal{X}_S^n$ we used a software package called PORTA which can, among other things, take a set of linear constraints which define a polytope and output an exhaustive list of its extreme points. PORTA is a free software package which was originally created by Thomas Christof but is now maintained by Andreas Löbel [21]. We gave PORTA the set of node equalities, cut constraints, and nonnegativity constraints which define $\mathcal{P}_S^n$ and the program produced the points of $\mathcal{X}_S^n$ for each $3 \leq n \leq 6$. Since PORTA uses floating point arithmetic to perform its calculations and our polytope is rational, we later verified our results with another software package called cddr+. This software package, like PORTA, is free and can find all the extreme points of a polytope given a description using linear constraints. It was created by Komei Fukuda [36] and can perform calculations using rational arithmetic, thereby eliminating any possible rounding errors.

Although we did exhaustively generate all the extreme points of $\mathcal{X}_S^n$ for each $3 \leq n \leq 6$ it is important later on to realize the following fact.

Proposition 4.3.1. *Let ϕ be a permutation of the nodes of V and let x be an extreme point of the ASEP on n nodes. Define $\phi(x)$ by*

$$(\phi(x))_{uv} = x_{\phi(u)\phi(v)}.$$

Then $\phi(x)$ is also an extreme point of the ASEP on n nodes. Furthermore, $Gap(\phi(x)) = Gap(x)$.

Proof. By Theorem 2.3.4, x is the unique solution to a set of equalities. By applying the permutation ϕ to these equalities we see that $\phi(x)$ is the unique solution to this set of permuted equalities. Hence $\phi(x)$ is an extreme point of the ASEP on n nodes. Also, if c is a metric cost function which attains the integrality gap, Gap(x), and t is the characteristic vector of a minimum cost tour of K_n with respect to c then $\text{Gap}(\phi(x)) = \frac{\phi(c)\phi(t)}{\phi(c)\phi(x)}$. But since ϕ is a permutation of V, $\phi(c)\phi(t) = ct$ and $\phi(c)\phi(x) = cx$. Therefore, $\text{Gap}(\phi(x)) = \text{Gap}(x)$. □

We say that two extreme points, x and x', are *isomorphic* if there is a permutation, ϕ, of V such that $x' = \phi(x)$. Clearly, the relation of two extreme points being isomorphic is an equivalence relation so we can partition $\mathcal{X}_S^n$ into equivalence classes. From Proposition 4.3.1, we also see that all of the extreme points in the same equivalence class have the same integrality gap. Thus when computing Gap_n we only need to compute Gap(x) for a single representative, x, of each equivalence class. In order to accomplish this, we used a software package called NAUTY created by Brendan McKay [63] to remove multiple representatives from the same equivalence classes. The results are summarized in Table 4.1.

n	Number of Extreme Points	Number of Equivalence Classes	Gap_n
3	2	1	1
4	12	2	6/5
5	384	5	5/4
6	57720	90	4/3

Table 4.1: Integrality gap for $3 \leq n \leq 6$

Interestingly enough, the integrality gap for each value of $4 \leq n \leq 6$ is attained by a unique (up to isomorphism) half-integer extreme point. The cost vectors which accompany them, however, are not unique. We display these important extreme points in Figures 4.3, 4.4, and 4.5. The support digraph for each extreme point is given and they are all half-integer points with no 1-arcs so the x-value of all arcs in the support digraph is $\frac{1}{2}$. The number appearing on each arc is a cost for the arc. The costs for arcs not shown is given by the metric completion and the costs are scaled so that they are all integer.

We see in the next section that the methods used to compute the integrality gap for $3 \leq n \leq 6$ are not practical for $n = 7$. Hence we need to further investigate the properties of the ASEP in order to continue our work.

4.4 Generating $\mathcal{X}_S^7$

We were unable to directly generate $\mathcal{X}_S^7$ using PORTA. When we did make our initial attempt, the program ran for over a week before we terminated it. We needed to find

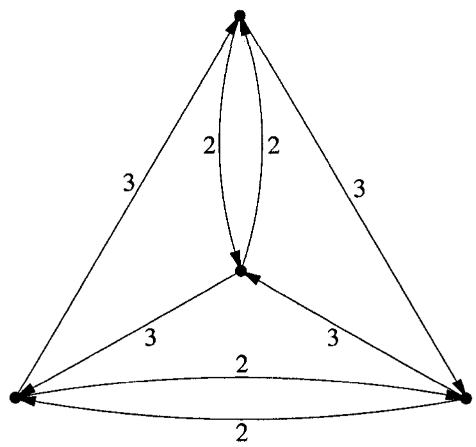

Figure 4.3: Largest integrality gap for $n = 4$

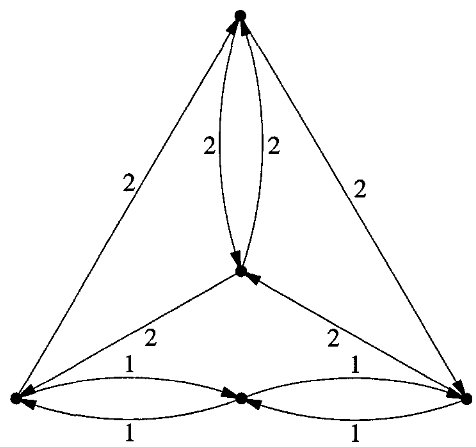

Figure 4.4: Largest integrality gap for $n = 5$

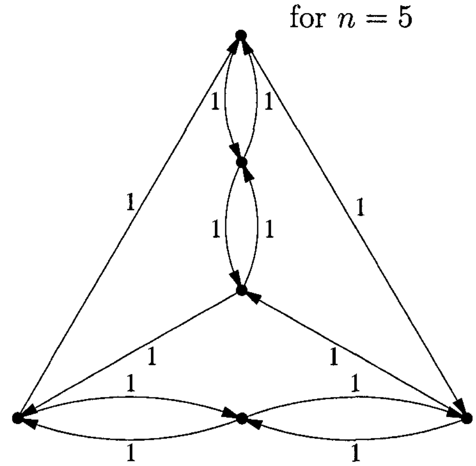

Figure 4.5: Largest integrality gap for $n = 6$

a way to reduce the number of calculations performed by PORTA. This lead us to the following observation.

Lemma 4.4.1. *Let x be an extreme point of the ASEP on n nodes and let $D(x)$ be the support digraph of x. Then $D(x)$ has two nodes which each have outdegree at most two.*

Proof. By Theorem 3.2.4, we know that $D(x)$ has at most $3n - 4$ arcs. If all the nodes of $D(x)$ have outdegree at least three then $D(x)$ has at least $3n$ arcs which is a contradiction. If all but one of the nodes of $D(x)$ have outdegree at least three then $D(x)$ has at least $3(n-1) = 3n-3$ arcs which again is a contradiction. Thus $D(x)$ must have at least two nodes, each of which having outdegree at most two. □

As a result, we chose two nodes of K_7 and set their outdegree to be at most two. Considering the way these two nodes could interact and removing any isomorphic cases, we were left with 41 different cases. Once we had chosen which arcs might be present with their tails at the two specified nodes, we knew that we could eliminate all other arcs in the support digraph with their tails at these nodes. This enabled us, in each of the 41 cases, to eliminate at least 10 variables from the linear inequality description of the polytope. Thus, instead of giving the description of one polytope to PORTA, we gave 41 simpler polytopes. PORTA then found the resulting extreme points for each of the 41 cases and these were combined into a large list. As a final step, we removed all but one extreme point in each isomorphism equivalence class. As a result, we found 3748 non-isomorphic extreme points. As a check, we verified our results using cddr+. Surprisingly (since the integrality gaps strictly increased for $3 \leq n \leq 6$ and they also strictly increase for the integrality gap of the STSP for all known small values of n), we computed that

$$\text{Gap}_7 = \frac{4}{3}$$

and, unlike for $3 \leq n \leq 6$, there are five non-isomorphic extreme points which attain this integrality gap. These five extreme points are depicted in Figures 4.6, 4.7, 4.8, 4.9, and 4.10. Again the costs are listed for each arc. The thick arcs have an x-value of 1 and the thin arcs have an x-value of $\frac{1}{2}$.

Again, we need to change our tactics when generating all the non-isomorphic extreme points for $n = 8$. Unfortunately, we do not find them all.

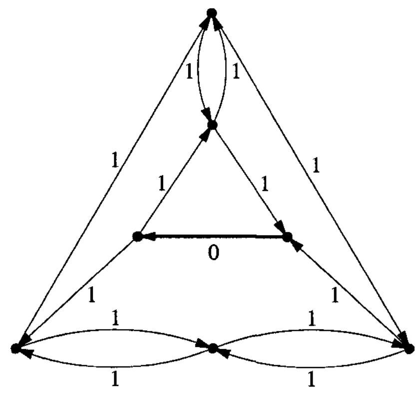

Figure 4.6: First extreme point for $n = 7$ with integrality gap 4/3

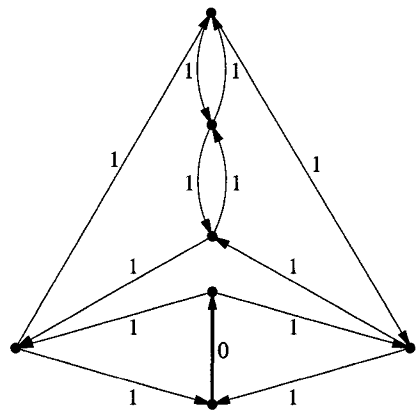

Figure 4.7: Second extreme point for $n = 7$ with integrality gap 4/3

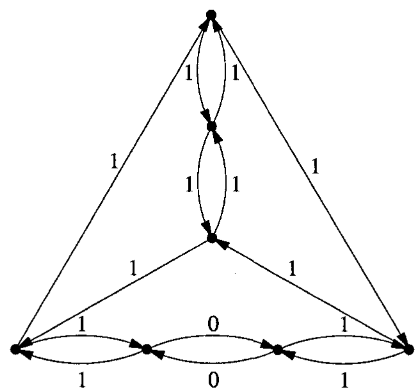

Figure 4.8: Third extreme point for $n = 7$ with integrality gap 4/3

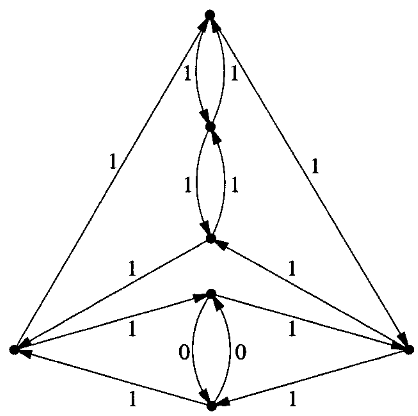

Figure 4.9: Fourth extreme point for $n = 7$ with integrality gap 4/3

4.5 Generating $\mathcal{X}_S^8$ and $\mathcal{X}_S^9$

Considering the length of time it took us to generate all the non-isomorphic extreme points of $\mathcal{X}_S^7$ (over 20 hours using cddr+ on a SunBlade 150 workstation, running sparc solaris 5.9 at 550 MHz with 640 MB of RAM) and the number of different cases

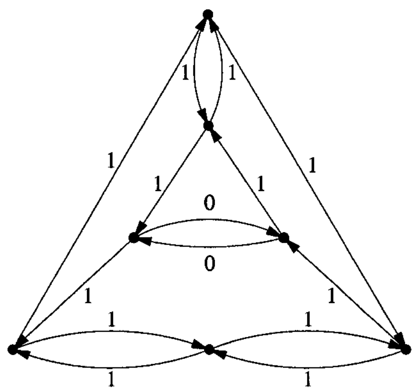

Figure 4.10: Fifth extreme point for $n = 7$ with integrality gap 4/3

considered, we abandoned that method of generating the extreme points when $n = 8$ and $n = 9$. Since we also noted that all the extreme points which attained Gap_n for each $3 \leq n \leq 7$ were half-integer, we attempted instead to generate all of the half-integer extreme points of $\mathcal{X}_S^8$ and $\mathcal{X}_S^9$.

In order to accomplish this, we note that the support digraph of every half-integer extreme point has the property that every node has indegree and outdegree at most 2. Furthermore, if a node is the tail of only one arc then this arc must be a 1-arc and so the head of this arc must have indegree 1. Similarly if a node is the head of only one arc then the tail of this arc must have outdegree 1. By duplicating any such arc, we get a directed multigraph where the indegree and outdegree of every node is exactly 2. Since the outdegree is the same as the indegree of every node, a well-known result states that there exists a closed directed trail in any such digraph which visits each node at least once and each arc in the directed multigraph exactly once. Since each node has outdegree and indegree 2, any such trail must visit each node exactly twice.

We then generated all such trails by generating all possible sequences of the nodes which contain each node exactly twice (but not the same node twice in a row). We removed any extra isomorphic copies and used PORTA to then find all the extreme points with the given support digraph. Certainly every non-isomorphic half-integer extreme point was generated, but there were also some other non-half-integer extreme points that were generated as well. In this way we found 1119 non-isomorphic extreme points of $\mathcal{X}_S^8$ and 10863 non-isomorphic extreme points of $\mathcal{X}_S^9$. Of all these extreme points generated, we found 43 extreme points for $n = 8$ which had an integrality gap of $\frac{4}{3}$. Surprisingly (since this had not occurred for smaller values of n), one of these extreme points is not half-integer and is depicted in Figure 4.11. In this figure, the thick arcs denote an x-value of $\frac{3}{4}$ while the thin arcs denote an x-value of $\frac{1}{4}$. For $n = 9$ we discovered that an integrality gap of $\frac{11}{8}$ is attained by a unique half-integer extreme point which is depicted in Figure 4.12. This is the smallest (in terms of the number of nodes) known example of an extreme point which attains an integrality gap larger than 4/3.

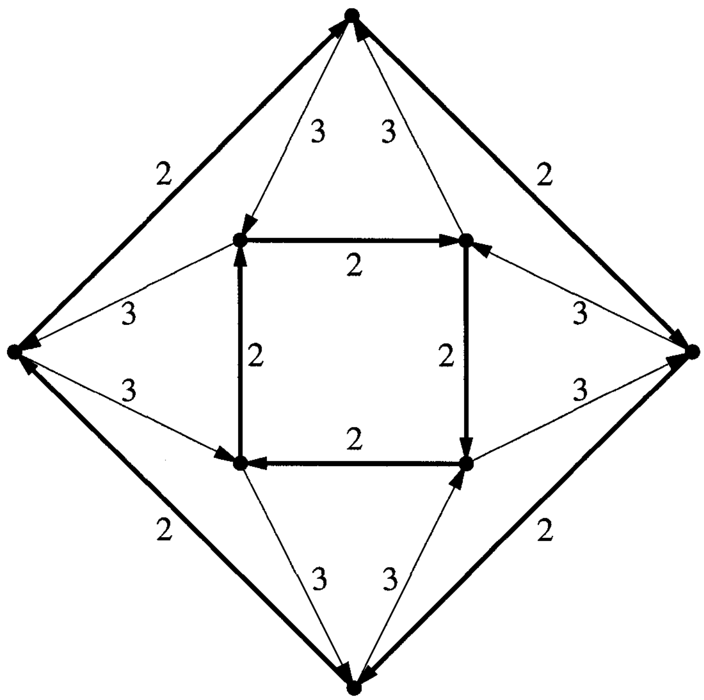

Figure 4.11: An interesting extreme point of $\mathcal{X}_S^8$

We continue our investigation into half-integer extreme points in the next chapter. Unfortunately, for $n \geq 10$, we do not find them all. We finish this chapter with a

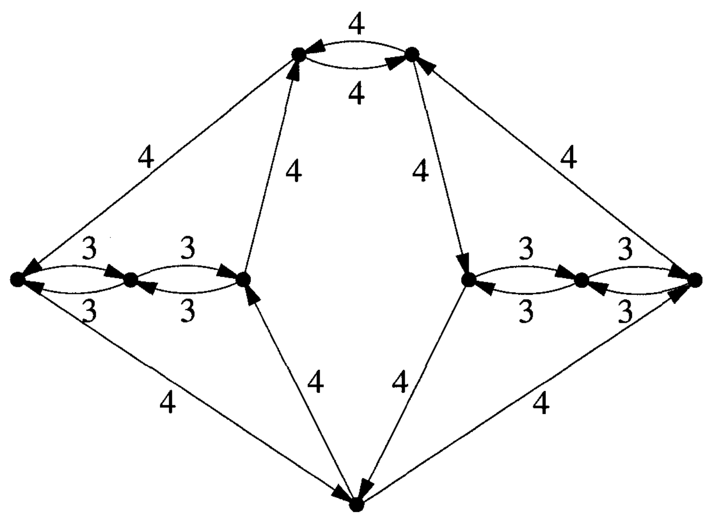

Figure 4.12: Best known integrality gap for $n = 9$

helpful theorem which eliminates the need to generate all the non-isomorphic extreme points when computing the integrality gap.

4.6 Inserting 1-arcs and the Integrality Gap

We show in this section the precise effect that inserting a 1-arc has on the integrality gap. In essence, we discover that inserting a 1-arc never increases the integrality gap and so in our search for extreme points with large integrality gaps, we can ignore extreme points which contain 1-arcs. We begin this section with a well-known proposition which allows us to restrict our attention to the arcs of the support digraph for a given extreme point.

Proposition 4.6.1. *Let x be an extreme point of the ASEP on n nodes and let c be a metric cost vector which attains the integrality gap $Gap(x)$. Let $c^{E(x)}$ be c restricted to the arcs of $E(x)$. Then the metric completion of $c^{E(x)}$ is also a metric cost vector which attains the integrality gap $Gap(x)$.*

Proof. Let c' be the metric completion of $c^{E(x)}$. Due to the cut constraints, we know that $D(x)$ is strongly connected and so, by Lemma 4.1.1, c' is metric. Since $c'_{uv} = c_{uv}$ for every $uv \in E(x)$, we have that $c'x = cx$. Furthermore, since c is metric $c' \geq c$, so for any tour T of K_n we have that $c'(T) \geq c(T)$. Let $T^\star$ be a minimum cost tour of K_n with respect to c'. Since $T^\star$ may not be a minimum cost tour with respect to c we have that

$$\begin{aligned} \text{Gap}(x) &\leq \frac{c(T^\star)}{cx} \\ &= \frac{c(T^\star)}{c'x} \\ &\leq \frac{c'(T^\star)}{c'x}. \end{aligned}$$

However, $T^\star$ is a minimum cost tour of K_n with respect to c' so $\text{Gap}(x) \geq \frac{c'(T^\star)}{c'x}$. Therefore, $\text{Gap}(x) = \frac{c'(T^\star)}{c'x}$. □

Theorem 4.6.2. *Let x be an extreme point on the ASEP on $n \geq 4$ nodes. If uv is a 1-arc of $E(x)$ then*

$$Gap(x \downarrow_w (\{u, v\})) \geq Gap(x).$$

Proof. Let $x' = x \downarrow_w (\{u, v\})$ and let V' be the node set of $D(x')$.

Let c be a metric cost vector which attains the integrality gap $\text{Gap}(x)$. By Proposition 4.6.1, we may assume that c_{ab} is the cost of a minimum cost (a, b)-dipath in $D(x)$ whenever $ab \notin E(x)$.

Define c' to be the following costs assigned to the arcs of K_{n-1}:

$$c'_{ab} = \begin{cases} c_{vb} + c_{uv} & \text{if } a = w \\ c_{au} & \text{if } b = w \\ c_{ab} & \text{otherwise.} \end{cases}$$

Clearly, $c' \geq 0$ so we just need to show that the triangle equalities hold. Let a, b, and r be three distinct nodes of V'. We consider four different cases.

Case 1: $a, b, r \in V' \backslash \{w\}$

$$\begin{aligned} c'_{ab} &= c_{ab} \\ &\leq c_{ar} + c_{rb} \\ &= c'_{ar} + c'_{rb} \end{aligned}$$

Case 2: $a = w$

$$\begin{aligned} c'_{ab} &= c_{vb} + c_{uv} \\ &\leq c_{rb} + c_{vr} + c_{uv} \\ &= c'_{rb} + c'_{wr} \\ &= c'_{rb} + c'_{ar} \end{aligned}$$

Case 3: $b = w$

$$\begin{aligned} c'_{ab} &= c_{au} \\ &\leq c_{ar} + c_{ru} \\ &= c'_{ar} + c'_{rw} \\ &= c'_{ar} + c'_{rb} \end{aligned}$$

Case 4: $r = w$

$$\begin{aligned} c'_{ab} &= c_{ab} \\ &\leq c_{au} + c_{ub} \\ &\leq c_{au} + c_{uv} + c_{vb} \\ &= c'_{aw} + c'_{wb} \\ &= c'_{ar} + c'_{rb} \end{aligned}$$

In all cases, we contradict the fact that c obeys the triangle inequalities. Therefore, c' is metric.

Now

$$\begin{aligned}
c'x' &= \sum_{\substack{a \in V' \\ b \in V'\setminus\{a\}}} c'_{ab}x'_{ab} \\
&= \sum_{b\in V'\setminus\{w\}} c'_{wb}x'_{wb} + \sum_{a\in V'\setminus\{w\}} c'_{aw}x'_{aw} + \sum_{\substack{a \in V'\setminus\{w\} \\ b \in V'\setminus\{a,w\}}} c'_{ab}x'_{ab} \\
&= \sum_{b\in V'\setminus\{w\}} (c_{vb}+c_{uv})x'_{wb} + \sum_{a\in V'\setminus\{w\}} c_{au}x'_{aw} + \sum_{\substack{a \in V'\setminus\{w\} \\ b \in V'\setminus\{a,w\}}} c_{ab}x'_{ab} \\
&= \sum_{b\in V\setminus\{u,v\}} (c_{vb}+c_{uv})x'_{wb} + \sum_{a\in V\setminus\{u,v\}} c_{au}x'_{aw} + \sum_{\substack{a \in V\setminus\{u,v\} \\ b \in V\setminus\{a,u,v\}}} c_{ab}x'_{ab} \\
&= \sum_{b\in V\setminus\{u,v\}} (c_{vb}+c_{uv})x_{vb} + \sum_{a\in V\setminus\{u,v\}} c_{au}x_{au} + \sum_{\substack{a \in V\setminus\{u,v\} \\ b \in V\setminus\{a,u,v\}}} c_{ab}x_{ab} \\
&= \sum_{b\in V\setminus\{u,v\}} c_{vb}x_{vb} + \sum_{b\in V\setminus\{u,v\}} c_{uv}x_{vb} + \sum_{a\in V\setminus\{u,v\}} c_{au}x_{au} + \sum_{\substack{a \in V\setminus\{u,v\} \\ b \in V\setminus\{a,u,v\}}} c_{ab}x_{ab} \\
&= \sum_{b\in V\setminus\{u,v\}} c_{vb}x_{vb} + c_{uv}\sum_{b\in V\setminus\{u,v\}} x_{vb} + \sum_{a\in V\setminus\{u,v\}} c_{au}x_{au} + \sum_{\substack{a \in V\setminus\{u,v\} \\ b \in V\setminus\{a,u,v\}}} c_{ab}x_{ab} \\
&= \sum_{b\in V\setminus\{u,v\}} c_{vb}x_{vb} + c_{uv}(1) + \sum_{a\in V\setminus\{u,v\}} c_{au}x_{au} + \sum_{\substack{a \in V\setminus\{u,v\} \\ b \in V\setminus\{a,u,v\}}} c_{ab}x_{ab} \\
&= c_{uv}x_{uv} + \sum_{\substack{a \in V\setminus\{u\} \\ b \in V\setminus\{a,v\}}} c_{ab}x_{ab} \\
&= cx.
\end{aligned}$$

Let T' be a minimum cost tour with respect to c'. Since c' may not attain the integrality gap for x' we have that

$$\mathrm{Gap}(x') \geq \frac{c'(T')}{c'x'}.$$

Now, let rw and ws be the unique arcs of T' which have their head and tail respectively at w. Define a new tour T on K_n where $T = (T' \backslash \{rw, ws\}) \cup \{ru, uv, vs\}$. Then

$$\begin{aligned} c'(T') &= \sum_{ab \in T'} c'_{ab} \\ &= c'_{rw} + c'_{ws} + \sum_{ab \in T' \backslash \{rw, ws\}} c'_{ab} \\ &= c_{ru} + c_{uv} + c_{vs} + \sum_{ab \in T' \backslash \{rw, ws\}} c_{ab} \\ &= \sum_{ab \in T} c_{ab} \\ &= c(T). \end{aligned}$$

However, T may not be a minimum cost tour with respect to c so we have

$$\mathrm{Gap}(x) \leq \frac{c(T)}{cx}.$$

Therefore,

$$\begin{aligned} \mathrm{Gap}(x') &\geq \frac{c'(T')}{c'x'} \\ &= \frac{c(T)}{cx} \\ &\geq \mathrm{Gap}(x). \end{aligned}$$

□

In the next chapter, we examine the impact of the 2-jack operation on the half-integer extreme points and their corresponding integrality gaps.

Chapter 5

Families of Half-integer Extreme Points

In the last chapter, we discuss the integrality gap for the metric ATSP. We then compute this integrality gap precisely for $4 \leq n \leq 7$. We also compute the integrality gap for half-integer extreme points for $n = 8$ and $n = 9$. We introduce two operations for generating new extreme points of the ASEP. In this chapter, we put these results together to generate new half-integer extreme points and compute their integrality gaps. Just as the numerical results in the previous chapters provide a lower bound on the integrality gap for the metric ATSP, our intention in this chapter is to explore the patterns of the extreme points that produce large integrality gaps and to establish a better lower bound on the integrality gap of the metric ATSP.

Carr and Vempala [16] conjectured that the integrality gap of the metric ATSP is at most $4/3$. The first family that we present in this chapter contains extreme points whose integrality gaps are arbitrarily close to $3/2$ and hence refutes this conjecture. We were not the only ones to find a counterexample to the conjecture. Charikar, Goemans, and Karloff [18] independently found a family of half-integer extreme points whose integrality gaps are arbitrarily close to 2. As a direct consequence, they show that the integrality gap of the metric ATSP is at least 2. We then use their idea of recursion to create a second family of half-integer extreme points which match the pattern (as outlined in the previous chapter) we see in our numerical results and whose integrality gaps approach 2. We finish the chapter by comparing our second family to that of Charikar, Goemans, and Karloff.

5.1 Inserting a 1-arc and the 2-jack Operations

In the last chapter, we see that if we have some half-integer extreme points of the ASEP on n nodes then we can use the 2-jack operation or insert a 1-arc to create new half-integer extreme points of the ASEP on $n+1$ nodes. In fact, we se that if we start with the unique (half-integer) extreme point of the ASEP on 4 nodes which attains the integrality gap then, by repeatedly applying the 2-jack operation, we can create a half-integer extreme point of $\mathcal{X}_S^n$ for each $5 \leq n \leq 9$ which attains the best known integrality gap.

Unfortunately, generating all the extreme points, or even all the half-integer extreme points, for $n \geq 10$ took too much time using the methods described in the previous chapter. We focus instead on generating half-integer extreme points using the 1-arc insertion and 2-jack operations. We see, from the example in Figure 3.8, that not every half-integer extreme point of the ASEP on $n+1$ nodes can be obtained via these operations applied to the half-integer points of $\mathcal{X}_S^n$. Hence, we are not able to generate all of the half-integer extreme points in this manner, but hopefully we can construct many of them. For each $4 \leq n \leq 9$, we take each of the non-isomorphic half-integer points points of $\mathcal{X}_S^n$ and apply the 1-arc insertion and 2-jack operations in all possible ways to obtain some half-integer points of $\mathcal{X}_S^{n+1}$. Since we do not have access to all the half-integer extreme points of the ASEP on 10 nodes, we apply the operations in all possible ways to those we generate from the half-integer points of $\mathcal{X}_S^9$. This gives partial results for $n = 11$. The results are shown in Table 5.1 where the headings denote the number of extreme points, the number of half-integer extreme points, and the number of half-integer extreme points we obtain in the above manner by applying the 1-arc insertion and 2-jack operations. The * serves as a reminder that the half-integer points of $\mathcal{X}_S^{11}$ obtained by the operations are created in a slightly different way (as described above). Notice that for each $5 \leq n \leq 9$ we are able to construct more than 90% of the half-integer extreme points via this method.

n	Extreme Points	Half-Integer	Obtained by Operations
4	2	2	-
5	5	4	4
6	90	21	19
7	3748	121	114
8		914	864
9		7653	7222
10			67513
11			595773*

Table 5.1: Half-integer extreme points obtained by the operations

Next, we consider the integrality gap of half-integer extreme points produced using these operations. As we se in Theorem 4.6.2, inserting a 1-arc can never increase the integrality gap. Hence, if we are trying to find half-integer extreme points with large integrality gaps, we can restrict our attention to the points obtained via the 2-jack operation. Since there are so many half-integer extreme points of the ASEP on 11 nodes generated by the operations, we are not able to apply the 2-jack operation in all possible ways to each one of them. Instead, for each value of n, $11 \leq n \leq 14$, we take the half-integer points of $\mathcal{X}_S^n$, that we know of, with the top several values of the integrality gap and we apply the 2-jack operation in all possible ways to these points to get some half-integer points of $\mathcal{X}_S^{n+1}$. This process generates many points which allows us to find a lower bound on Gap_n for each $11 \leq n \leq 15$. We summarize the results of our exploration in Table 5.2 where the rightmost column contains a lower bound on the value of Gap_n.

n	Lower Bound on Gap_n
11	10/7
12	56/39
13	13/9
14	100/69
15	16/11

Table 5.2: Best integrality gaps found

Let us define a pair of antiparallel arcs to be called a *digon* and let a *k-chain* be the digraph obtained by replacing each edge in an undirected open path of length k by a digon. A digon and a 3-chain are shown in Figure 5.1 and Figure 5.2 respectively. What is surprising is that, for each value of $11 \leq n \leq 15$, we find a unique half-integer extreme point which attains the largest known gap. Each one is obtained from the half-integer extreme point shown in Figure 5.3 by alternately performing the 2-jack operation at the nodes labelled a and b to extend the two 2-chains.

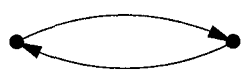

Figure 5.1: A digon

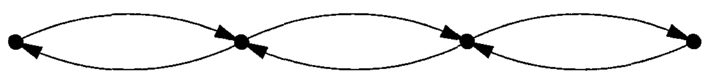

Figure 5.2: A 3-chain

We then focus our attention on extending the two 2-chains in Figure 5.3 to two k-chains for some $k \geq 3$ by repeatedly applying the 2-jack operation at the nodes labelled a and b. As a result, consider the half-integer extreme point of the ASEP shown in Figure 5.4. The values associated with each arc give the arc cost and the

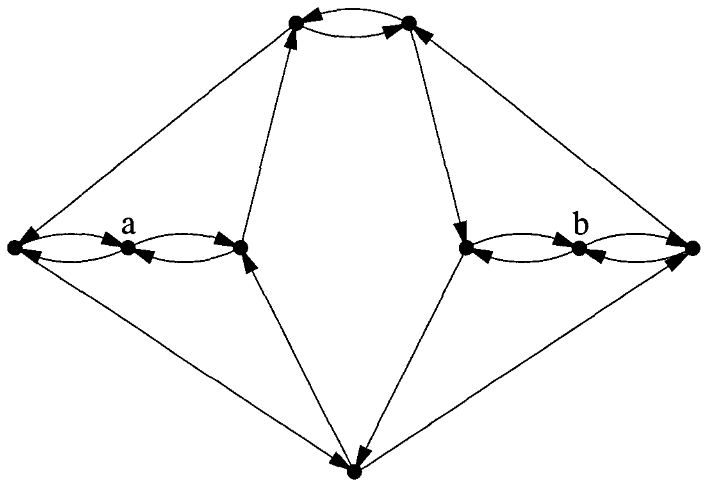

Figure 5.3: Best gap extreme point for $n = 9$

costs for arcs not shown are obtained from the metric completion. An example for the special case when $k = 4$ is shown in Figure 5.5. We can calculate that the total cost of this extreme point with respect to the given arc costs is $4k + 2$. In order to compute the optimum value of the ATSP with respect to the given arc costs, we need to introduce a slightly different problem, which is given in the next section.

5.2 Graphical Tours

A *graphical tour* of a strongly connected digraph is a closed directed walk which spans the digraph. Since a graphical tour is a directed walk, it may contain several copies of the same arc. Hence, we can think of a graphical tour as a directed multigraph. Note that this directed multigraph is Eulerian, and thus, by the directed version of a well-known result of Veblen [78], the directed multigraph can be decomposed into arc-disjoint dicycles. Thus, for the purposes of this thesis, we refer to a collection of dicycles whose union is a strongly connected directed multigraph as a *strongly connected collection of dicycles.* Hence a graphical tour is the union of a strongly connected collection of dicycles. Likewise, the union of a strongly connected collection

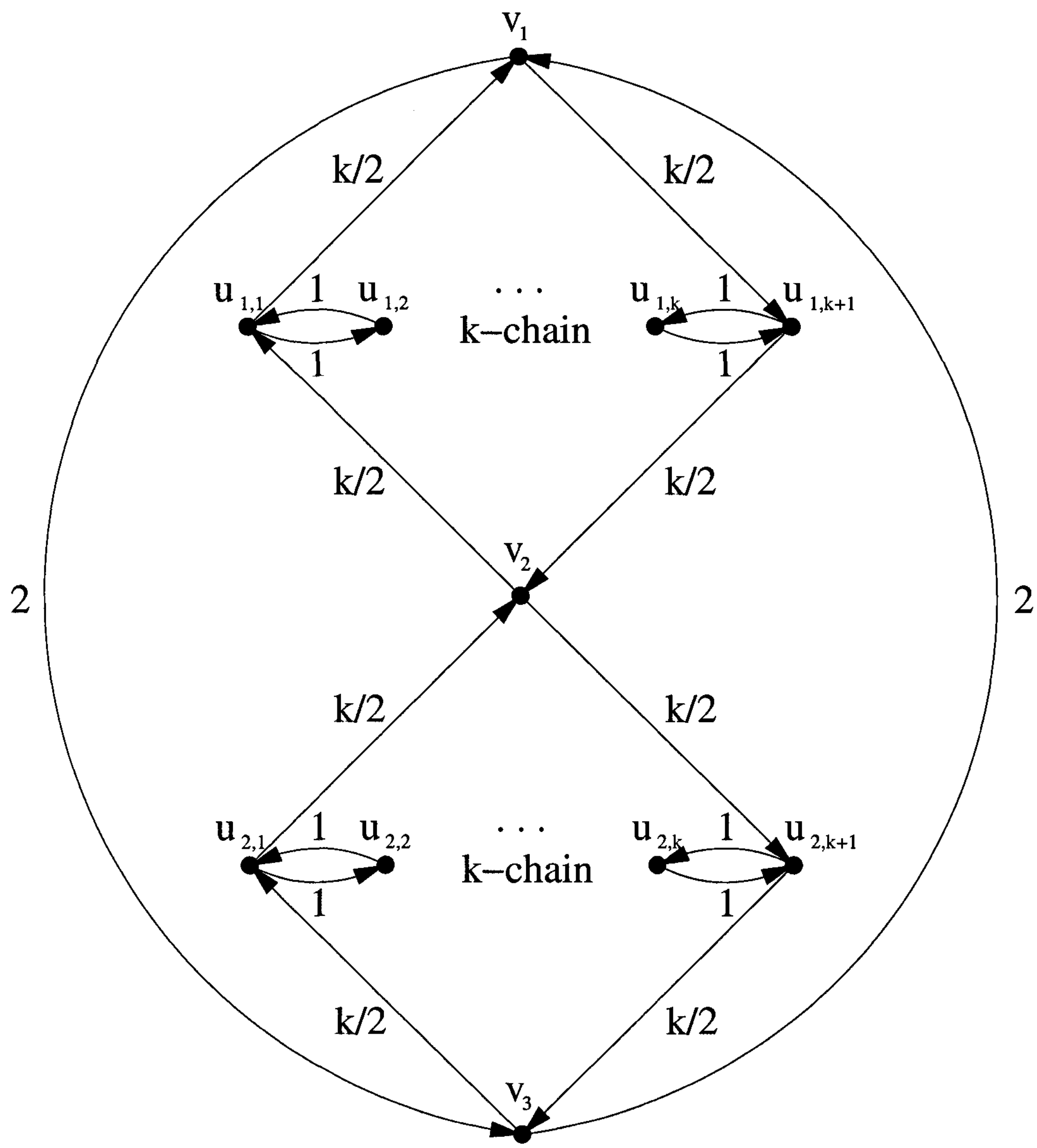

Figure 5.4: An extreme point with two k-chains

of dicycles in a digraph is an graphical tour. This characterization is used later in this section to find minimum cost graphical tours of the complete digraph K_n. The reason we are interested in graphical tours is stated in Lemma 5.2.1. This is a straightforward extension of the shortcutting procedure used for the metric STSP.

Lemma 5.2.1. *Let c be a set of metric costs defined on the arcs of the complete*

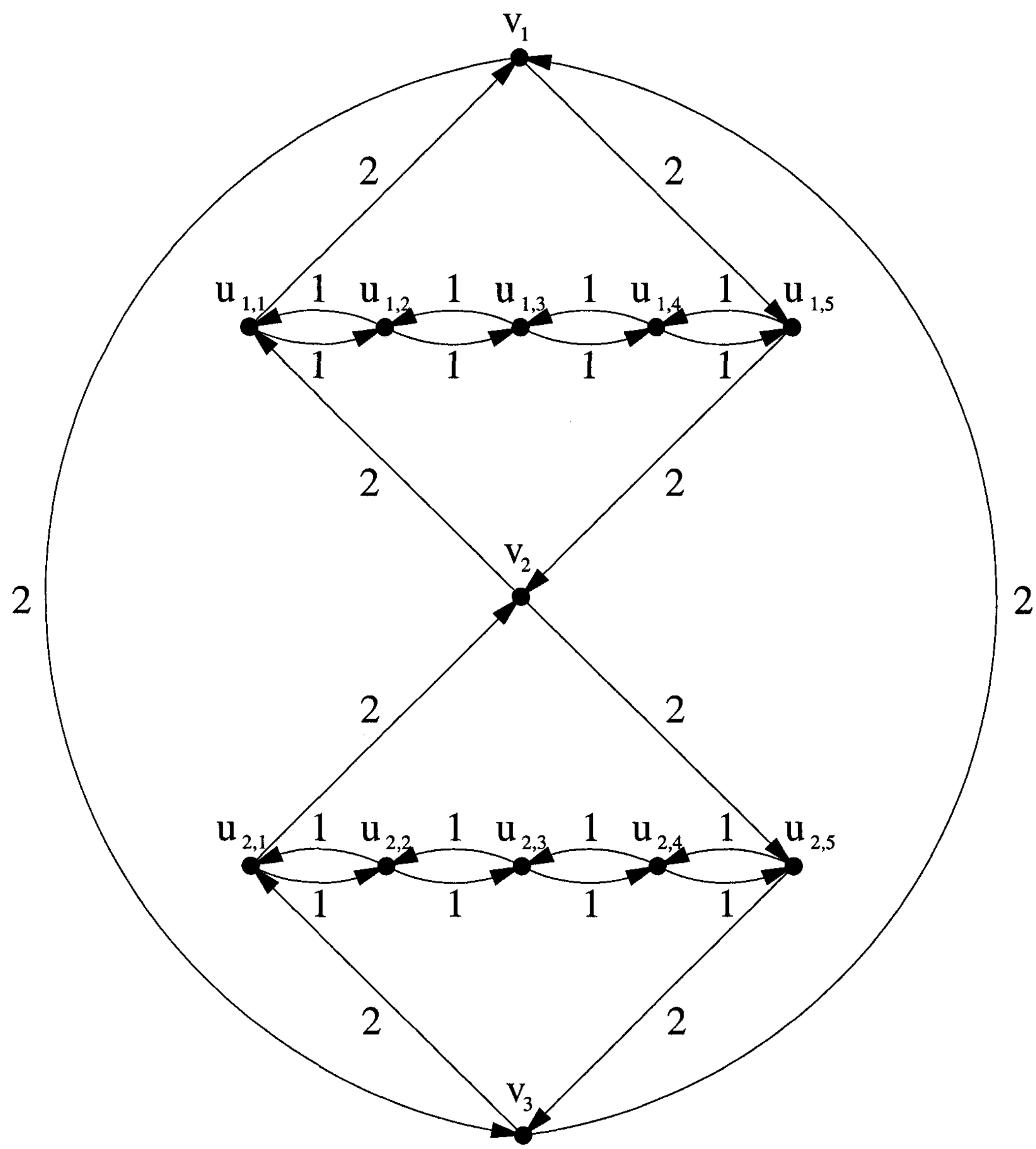

Figure 5.5: An extreme point with two 4-chains

digraph on n nodes. Then the cost of a minimum cost graphical tour with respect to c is the same as the optimal value of the ATSP with respect to c.

Proof. Any directed Hamilton cycle is also a graphical tour so the cost of a minimum cost graphical tour with respect to c is at most the optimal value of the ATSP with respect to c.

Now let Q be a minimum cost graphical tour with respect to c and let $v_0, v_1, \ldots, v_{k-1}, v_k$ be the nodes, in order, visited in Q where $v_k = v_0$. Since Q is a graphical tour, a node may appear several times in the above list. We can "shortcut" Q into a directed Hamilton cycle T by removing all but the first appearance of each node in the list (i.e. only retain the copy of the node with the smallest index). Let the new ordered list for T be $u_0, u_1, \ldots, u_{n-2}, u_{n-1}$. Since every directed walk contains a directed path between its endpoints, we know that Q contains arc-disjoint (u_i, u_{i+1})-dipaths for each $0 \leq i \leq n-1$. Let $P_0, \ldots, P_{n-1}$ be these dipaths. Since c is metric, we know that $c_{u_i u_{i+1}} \leq c(P_i)$ for each $0 \leq i \leq n-1$. Hence

$$c(Q) \geq \sum_{i=0}^{n-1} c_{u_i u_{i+1}}.$$

So we see that $c(Q) \geq c(T)$ and thus the cost of a minimum cost graphical tour with respect to c is at least the optimal value of the ATSP with respect to c. □

For our particular purposes, we use Lemma 5.2.1 to find the integrality gap of a specific ASEP extreme point in the following way. Given an extreme point x and a set of metric arc costs we assume without loss of generality, by Proposition 4.6.1, that c is the metric completion of the costs of the arcs of $E(x)$. Let T be an optimal solution to the ATSP with respect to c. If uv is an arc of T which is not an arc of $E(x)$, then we replace uv with a min-cost (u, v)-dipath in $D(x)$. By repeating this process for each such arc of $T - E(x)$ we get a set of arcs, Q, that comprise a directed closed walk which is strongly connected. Hence Q is a graphical tour of $D(x)$ and by our assumption of c being the metric completion of the costs of arcs in $E(x)$, we get that $c(Q) = c(T)$. Thus, by Lemma 5.2.1, we see that the problem of finding an optimal solution to the ATSP with respect to c is equivalent to finding a minimum cost graphical tour of $D(x)$ with respect to c. Furthermore, we can look for a minimum cost strongly connected collection of directed cycles of $D(x)$ to solve either problem. This is our approach in finding optimal solutions to the ATSP when computing the integrality gap of a given extreme point of the ASEP and is used throughout the remainder of this chapter.

5.3 A Family of Half-integer Extreme Points

Let us now return our attention to the ASEP extreme point, x^k, along with the set of arc costs, c^k, depicted in Figure 5.4. From Lemma 5.2.1, we can compute the cost of a minimum cost tour by finding a minimum cost strongly connected collection of dicycles of $D(x^k)$. In this section, we show that the integrality gap of c^k approaches $3/2$ as k approaches ∞. This refutes the conjecture of Carr and Vempala [16] that the integrality gap of the metric ATSP is at most $4/3$ (although it was also refuted independently in [18]). We finish this section by presenting an extreme point of the ASEP on 18 nodes that has an integrality gap of exactly $3/2$ (see Figure 5.7).

We start our exploration of x^k by examining the dicycles of $D(x^k)$. $D(x^k)$ has a limited number of different types of dicycles. Firstly, there are the digons in the k-chains. We call this type of dicycle a *link* and each link has cost 2. Secondly, there are the dicycles $(v_3, u_{2,1}, v_2, u_{1,1}, v_1)$ and $(v_1, u_{1,k+1}, v_2, u_{2,k+1}, v_3)$ which each have cost $2k+2$ and we call *dicycles of type A*. Thirdly, there are the dicycles $(v_i, u_{i,k+1}, v_{i+1}, u_{i,1})$ for each of $i = 1, 2$, which we call *dicycles of type B* and they each have cost $2k$. Fourthly, there are dicycles $(v_i, u_{i,k+1}, u_{i,k}, \ldots, u_{i,2}, u_{i,1})$ and $(v_{i+1}, u_{i,1}, u_{i,2}, \ldots, u_{i,k}, u_{i,k+1})$ for each of $i = 1, 2$, which we call *dicycles of type C* and they each have cost $2k$. Lastly, we have the digon (v_1, v_3) which has cost 4. Figure 5.6 depicts examples of a dicycle of type A, a dicycle of type B and a dicycle of type C. In each case, the dicycle in question is the directed face containing the label. We now have the language to more simply compute the optimal value of the ATSP with respect to c^k.

Theorem 5.3.1. *For each $k \geq 3$, we have ATSP(c^k) $= 6k + 2$.*

Proof. We build a collection of dicycles, $\mathcal{C}$, of $D(x^k)$ of minimum cost which is strongly connected. First, we note that there is no need to have both dicycles of type A in $\mathcal{C}$ since the union of these dicycles covers the same nodes as the two dicycles of type

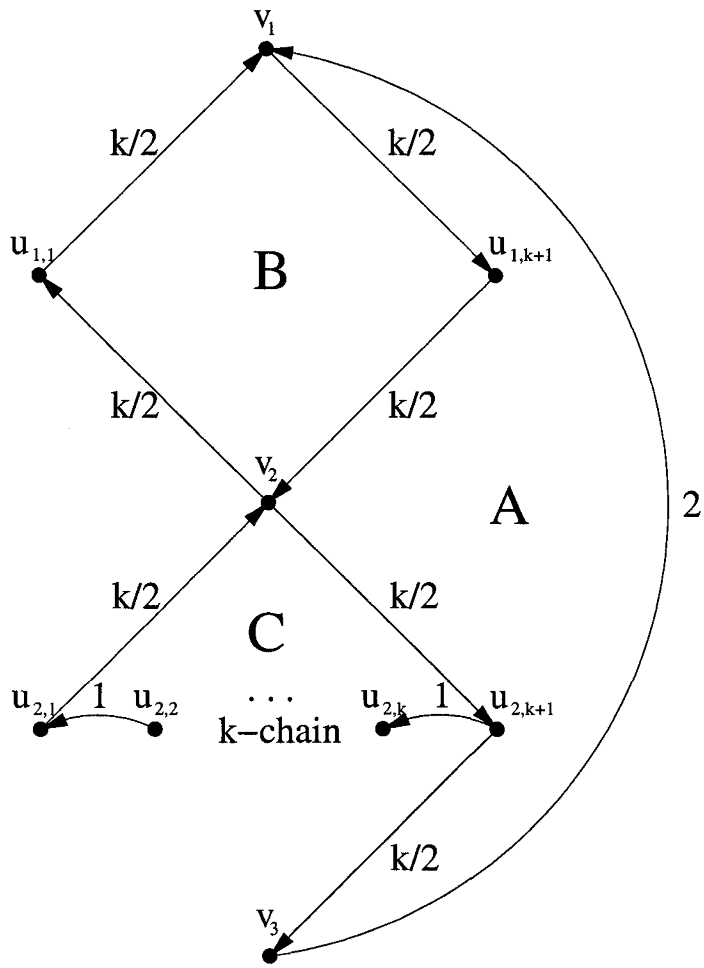

Figure 5.6: An extreme point with two k-chains

B whereas the two dicycles of type B have a lower total cost than those of type A. Hence we may assume that $\mathcal{C}$ contains at most one dicycle of type A.

Case 1: $\mathcal{C}$ contains exactly one dicycle of type A.

By symmetry, we may assume that $\mathcal{C}$ contains the dicycle $(v_1, u_{1,k+1}, v_2, u_{2,k+1}, v_3)$. Then $u_{1,1}$ must be covered by $(v_1, u_{1,k+1}, v_2, u_{1,1})$, $(v_1, u_{1,k+1}, u_{1,k}, \ldots, u_{1,2}, u_{1,1})$, $(v_2, u_{1,1}, u_{1,2}, \ldots, u_{1,k}, u_{1,k+1})$, or the links of the k-chain with endpoints $u_{1,1}$ and $u_{1,k+1}$. Each one of these options has a total cost of $2k$ but the dicycles of type C and the k-chain have the added benefit of covering the nodes $u_{1,2}, \ldots, u_{1,k}$ as well. Hence we may assume that $\mathcal{C}$ contains the links of the k-chain with endpoints $u_{1,1}$ and

$u_{1,k+1}$. By symmetry, we may also assume that $\mathcal{C}$ contains the links of the k-chain with endpoints $u_{2,1}$ and $u_{2,k+1}$. Hence we have built a strongly connected collection of dicycles which cover all the nodes of $D(x^k)$. The total cost of this collection is $6k+2$. This completes Case 1.

Case 2: $\mathcal{C}$ contains no dicycle of type A and at least one dicycle of type C.

Now, suppose that $\mathcal{C}$ contained both the dicycle $(v_i, u_{i,k+1}, u_{i,k}, \ldots, u_{i,2}, u_{i,1})$ and $(v_{i+1}, u_{i,1}, u_{i,2}, \ldots, u_{i,k}, u_{i,k+1})$ for $i = 1$ or $i = 2$. The total cost of these two dicycles of type C is $4k$. However, we could cover exactly the same nodes with the dicycle $(v_i, u_{i,k+1}, v_{i+1}, u_{i,1})$ and all but one of the links of the k-chain with endpoints $u_{i,1}$ and $u_{i,k+1}$. The latter subset of dicycles has a total cost of $4k-2$ and so the former subset of dicycles cannot be in our minimum cost collection of dicycles, $\mathcal{C}$. Similarly, if $\mathcal{C}$ contains both $(v_i, u_{i,k+1}, v_{i+1}, u_{i,1})$ and, either $(v_i, u_{i,k+1}, u_{i,k}, \ldots, u_{i,2}, u_{i,1})$ or $(v_{i+1}, u_{i,1}, u_{i,2}, \ldots, u_{i,k}, u_{i,k+1})$ for $i = 1, 2$, then the dicycle of type C can be replaced with all but one of the links of the k-chain with endpoints $u_{i,1}$ and $u_{i,k+1}$. Again the former subset of dicycles has cost $4k$ whereas the latter has cost $4k-2$. This contradicts the fact that $\mathcal{C}$ is a minimum cost collection of dicycles. Thus we see that there is precisely one dicycle of type B or C which covers the nodes $u_{1,1}$ and $u_{1,k+1}$. The same is true of $u_{2,1}$ and $u_{2,k+1}$.

Notice that there are precisely two $(v_1, u_{1,k+1})$-dipaths in $D(x^k)$ - namely $(v_1, u_{1,k+1})$ and $(v_1, v_3, u_{2,1}, v_2, u_{1,1}, u_{1,2}, \ldots, u_{1,k}, u_{1,k+1})$. Since the collection of dicycles in $\mathcal{C}$ must be strongly connected, if $v_1 u_{1,k+1}$ is not an arc contained in a dicycle of $\mathcal{C}$ then each of $v_1 v_3$, $v_3 u_{2,1}$, $u_{2,1} v_2$, and $v_2 u_{1,1}$ must be contained in the union of the dicycles of $\mathcal{C}$. In other words, if $\mathcal{C}$ contains the dicycle $(v_2, u_{1,1}, u_{1,2}, \ldots, u_{1,k}, u_{1,k+1})$ then it must also contain the dicycle $(v_2, u_{2,k+1}, v_3, u_{2,1})$ and the digon (v_1, v_3). In this situation, the only nodes which are not covered by these dicycles are $u_{2,2}, \ldots, u_{2,k}$. Since $(v_2, u_{2,k+1}, v_3, u_{2,1})$ is a dicycle of $\mathcal{C}$, we must cover $u_{2,2}, \ldots, u_{2,k}$ using links from the k-chain with endpoints $u_{2,1}$ and $u_{2,k+1}$. However, we only need $k-1$ of the links of this k-chain in order to cover the nodes and have a strongly connected collection of dicycles. A similar argument can be made if $\mathcal{C}$ contains any dicycle of type C. Hence, if $\mathcal{C}$ contains a dicycle of type C then it contains exactly one dicycle of type C, exactly one dicycle of type B, the digon (v_1, v_3), and $k-1$ links from the appropriate k-chain. In any case, the total cost of $\mathcal{C}$ is $2k + 2k + 4 + 2(k-1) = 6k + 2$. This completes Case 2.

Case 3: $\mathcal{C}$ contains no dicycle of type A and no dicycle of type C.

As discussed in Case 2, if $\mathcal{C}$ contains no dicycle of type A and no dicycle of type C then it must contain both dicycles of type B. In this case, there is no need for the digon (v_1, v_3) to be present in $\mathcal{C}$. The only other dicycles that can possibly be in $\mathcal{C}$ are the links of the two k-chains. In order to cover all the nodes of $D(x^k)$ and have a strongly connected set of dicycles we must use at least all but one of the links in each k-chain. This results in a minimum cost strongly connected collection of dicycles. The total cost is $2k + 2k + (2k-2) + (2k-2) = 8k - 4$. This completes Case 3.

Thus, we see from the three cases that the minimum cost of a strongly connected collection of dicycles of $D(x^k)$ is either $6k+2$ or $8k-4$. Since $k \geq 3$ we have that $6k + 2 \leq 8k - 4$. Therefore, the cost of $\mathcal{C}$ is $6k + 2$. Hence, by Lemma 5.2.1, $\text{ATSP}(c^k) = 6k + 2$. □

Corollary 5.3.2. *For every $k \geq 3$,*

$$Gap(c^k) = \frac{6k+2}{4k+2}.$$

Proof. As we see in Theorem 5.3.1, $\text{ATSP}(c^k) = 6k + 2$. Now, we construct a dual solution to the LP relaxation of the ATSP with respect to c^k to show that x^k is optimal with respect to c^k. Consider

$$y_v^{\text{out}} = \begin{cases} 1 & \text{if } v = v_1 \text{ or } v = v_3 \\ (k-1)/2 & \text{if } v = v_2 \\ 1/2 & \text{otherwise} \end{cases}$$

$$y_v^{\text{in}} = \begin{cases} 1 & \text{if } v = v_1 \text{ or } v = v_3 \\ (k-1)/2 & \text{if } v = v_2 \\ 1/2 & \text{otherwise} \end{cases}$$

$$d_S = \begin{cases} (k-3)/2 & \text{if } S = \{v_1, v_3\} \text{ or } S = V \backslash \{v_1, v_3\} \\ 0 & \text{otherwise} \end{cases} .$$

This choice of values for the variables satisfies the conditions of Theorem 4.2.4 and so x^k is an optimal solution of the LP. Hence

$$\text{ATSP}_{LP}(c^k) = c^k x^k = 4k + 2$$

and the result follows. □

We can see from Corollary 5.3.2 that, by choosing k large enough, we can achieve an integrality gap arbitrarily close (but smaller than) 3/2. We are, however, able to find extreme points that have integrality gaps of at least 3/2. The example we find with the smallest number of nodes is obtained by identifying the nodes u and v of Figure 5.7. This extreme point is half-integer with no 1-arcs and the costs for all the support digraph arcs are given.

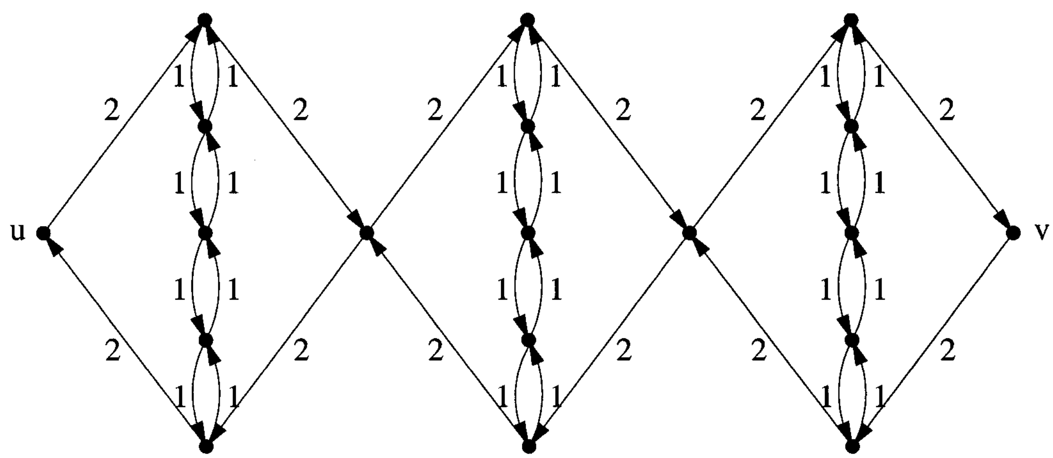

Figure 5.7: An extreme point with a gap of 3/2

In this section, we explore a generalization of the half-integer extreme points of the ASEP which have large integrality gaps. It was our hope to use this generalization to uncover other half-integer extreme points which have even larger integrality gaps. In the next section, we use recursion to generalize the family presented in this section to attain integrality gaps which are arbitrarily close to 2.

5.4 A Recursive Family of Half-integer Extreme Points

We see in the last section a family of extreme points, based on those half-integer extreme points which are known to attain large integrality gaps, whose integrality gaps could be made to be arbitrarily close to 3/2. In this section, we use that previous family as a base to recursively construct a new family of half-integer extreme points of the ASEP whose integrality gaps are arbitrarily close to 2.

Next, we introduce a family of digraphs, G_r^k, and we state some results about minimum-cost strongly connected collections of dicycles subject to certain properties. Although this family is not the primary focus of this section, these results help us to construct the new family of half-integer extreme points, x_r^k, which are of great interest in this section.

Now consider the digraph shown in Figure 5.8 which we call G_0^k where $k \geq 4$. This digraph consists of a digon (v_0, v_k), k directed 4-cycles which we call squares, and a k-chain in each square $(v_{i-1}, u_{i,k}, v_i, u_{i,0})$ with endpoints at $u_{i,0}$ and $u_{i,k}$ for each $1 \leq i \leq k$. The cost for each arc of G_0^k is given.

There are several different types of dicycles of G_0^k and we use notation similar to that in Theorem 5.3.1 to describe them. Firstly, there are the dicycles $(v_0, u_{1,k}, v_1, u_{2,k}, v_2, \ldots, v_{k-2}, u_{k-1,k}, v_{k-1}, u_{k,k}, v_k)$ and $(v_k, u_{k,0}, v_{k-1}, u_{k-1,0}, v_{k-2}, \ldots, v_2, u_{2,0}, v_1, u_{1,0}, v_0)$ which we call *dicycles of type A*. These dicycles have cost $2k^2$ each. Secondly, there are the squares $(v_{i-1}, u_{i,k}, v_i, u_{i,0})$

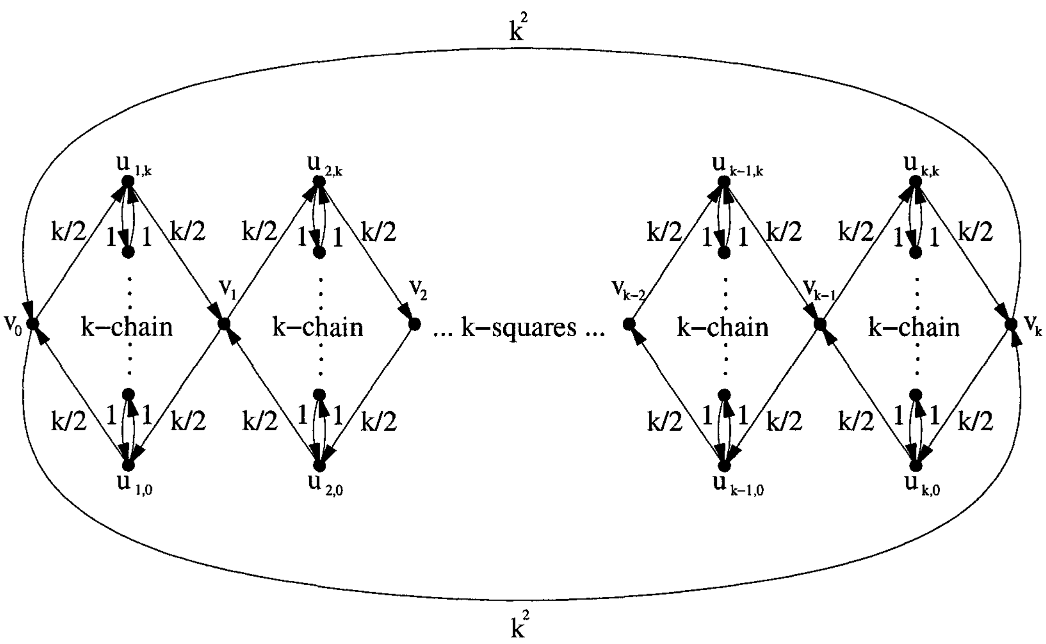

Figure 5.8: The digraph G_0^k

for each $1 \leq i \leq k$ which each have cost $2k$. Thirdly, there are the dicycles of the form $(v_{i-1}, u_{i,k}, u_{i,k-1}, \ldots, u_{i,1}, u_{i,0})$ and $(v_i, u_{i,0}, u_{i,1}, \ldots, u_{i,k-1}, u_{i,k})$ for each $1 \leq i \leq k$. We call them *dicycles of type C* and they each have cost $2k$. Fourthly, we have the links in the k-chains which each have cost 2. Lastly, we have the digon (v_0, v_k) which has cost $2k^2$.

We define q_0^k to be the minimum cost of a strongly connected collection of dicycles of G_0^k which contain at least one dicycle of type A. We define m_0^k to be the minimum cost of a strongly connected collection of dicycles of G_0^k which contain the digon (v_0, v_k).

Theorem 5.4.1. *For all $k \geq 4$ we have $q_0^k = 4k^2$.*

Proof. Assume, without loss of generality, that our collection of dicycles contains the dicycle of type A $(v_0, u_{1,k}, v_1, u_{2,k}, v_2, \ldots, v_{k-2}, u_{k-1,k}, v_{k-1}, u_{k,k}, v_k)$. Notice that there is no need to have the other dicycle of type A since we could replace it with all k squares, which also have a total cost of $2k^2$, and collectively cover all the same nodes of G_0^k. There is no need to have the digon (v_0, v_k) in our collection since the nodes v_0 and v_k are already covered by our dicycle of type A. Lastly, there is no need to have any square or dicycle of type C, since we could replace it with the appropriate k-chain, which also has cost $2k$, and still cover the same nodes left uncovered by the dicycle of type A. Furthermore, every link in each k-chain must be present in our collection in order for our collection to be strongly connected. Therefore, our collection of dicycles contains precisely a dicycle of type A and all k k-chains. The total cost of this collection is hence $2k^2 + k(2k) = 4k^2$. □

Theorem 5.4.2. *For all $k \geq 4$ we have $m_0^k = 6k^2 - 4k + 2$.*

Proof. Let $\mathcal{C}$ be a cheapest strongly connected collection of dicycles of G_0^k which contains the digon (v_0, v_k). Notice that there is no need to have a dicycle of type A in $\mathcal{C}$ since we can replace it with all k squares, which also have a total cost of $2k^2$, with the added benefit of covering many more nodes of G_0^k. Similar to what was noted in Theorem 5.3.1, there is no need to have either both dicycles of type C, $(v_{i-1}, u_{i,k}, u_{i,k-1}, \ldots, u_{i,1}, u_{i,0})$ and $(v_i, u_{i,0}, u_{i,1}, \ldots, u_{i,k-1}, u_{i,k})$, or the square $(v_{i-1}, u_{i,k}, v_i, u_{i,0})$ and one of these dicycles of type C for any $1 \leq i \leq k$ since, in either case, we can replace these dicycles with the square $(v_{i-1}, u_{i,k}, v_i, u_{i,0})$ and all but one of the links in the k-chain with endpoints $u_{i,0}$ and $u_{i,k}$. This replacement covers all the same nodes, but has total cost $4k - 2$ rather than $4k$.

Hence, if there is a dicycle of type C in $\mathcal{C}$, say $(v_{i-1}, u_{i,k}, u_{i,k-1}, \ldots, u_{i,1}, u_{i,0})$, then arcs $u_{i,k}v_i$ and $v_iu_{i,0}$ are not contained in any dicycle of $\mathcal{C}$. Now, there are precisely two (v_{i-1}, v_i)-dipaths in G_0^k – namely $(v_{i-1}, u_{i,k}, v_i)$ and $(v_{i-1}, u_{i-1,0}, v_{i-2}, \ldots, v_0, v_k, u_{k,0}, v_{k-1}, \ldots, v_i)$. Since $u_{i,k}v_i$ is not an arc in any dicycle of $\mathcal{C}$ and the collection of dicycles is strongly connected, all of the arcs in the dipath $(v_{i-1}, u_{i-1,0}, v_{i-2}, \ldots, v_0, v_k, u_{k,0}, v_{k-1}, \ldots, v_i)$ must be contained in the dicycles of $\mathcal{C}$. Thus, $\mathcal{C}$ cannot contain any more dicycles of type C. By similar arguments, we can see that $\mathcal{C}$ contains at most one dicycle of type C.

In order to cover all the nodes $v_1, \ldots, v_{k-1}$ and for $\mathcal{C}$ to be strongly connected, we see that either $\mathcal{C}$ contains all k squares or $\mathcal{C}$ contains $k-1$ squares and one dicycle of type C. If $\mathcal{C}$ contains the square $(v_{i-1}, u_{i,k}, v_i, u_{i,0})$ for some $1 \le i \le k$, then in order to cover the nodes $u_{i,1}, \dots, u_{i,k-1}$ we use all but one of the links in the k-chain with endpoints $u_{i,0}$ and $u_{i,k}$. Notice that a square and $k-1$ links have a total cost of $4k-2$ whereas a dicycle of type C has a total cost of $2k$. Since $\mathcal{C}$ has minimum cost, we conclude that $\mathcal{C}$ must consist of the digon (v_0, v_k), a dicycle of type C, $k-1$ squares, and $(k-1)(k-1)$ links from the k-chains with their endpoints in the squares in our collection. Therefore, the total cost of $\mathcal{C}$ is $2k^2+2k+(k-1)(2k)+(k-1)(k-1)(2) = 6k^2-4k+2$ as required. □

We now recursively define a family of digraphs with G_0^k as our base case. Digraph G_r^k for $r \ge 1$ is depicted in Figure 5.9. Here $H_{r-1}^k = G_{r-1}^k - \{v_0v_k, v_kv_0\}$ and to construct G_r^k, we take G_0^k and replace each k-chain with a copy of H_{r-1}^k. We identify the nodes v_0 and v_k in the copy of H_{r-1}^k with $u_{i,0}$ and $u_{i,k}$ in G_0^k respectively. The costs of the arcs on the copies of H_{r-1}^k are the same as they would be in G_{r-1}^k. The costs of all other arcs are given in the diagram.

Similar to the case of G_0^k, the dicycles of G_r^k are the digon (v_0, v_k), the squares, and the dicycles of type A defined as they were for G_0^k. These dicycles have costs $2k^{r+2}$, $2k^{r+1}$, and $2k^{r+2}$, respectively. In addition, we have the dicycles of type C which consist of either the arcs $u_{i,k}v_i$ and $v_iu_{i,0}$ along with unique (v_0, v_k)-dipath of H_{r-1}^k for some $1 \le i \le k$ or else the arcs $u_{i,0}v_{i-1}$ and $v_{i-1}u_{i,k}$ along with the unique (v_k, v_0)-dipath of H_{r-1}^k for some $1 \le i \le k$. In either case, the total cost of the dicycle of type C is $2k^{r+1}$. Finally, the only other dicycles of G_r^k are the dicycles contained within each of the copies of H_{r-1}^k. These are precisely the dicycles of G_{r-1}^k less the digon (v_0, v_k) and the two dicycles of type A. We define q_r^k to be the cost of a minimum cost strongly connected collection of dicycles of G_r^k which contains at least one dicycle of type A. We also define m_r^k to be the cost of a minimum cost strongly connected collection of dicycles of G_r^k which contains the digon (v_0, v_k).

Theorem 5.4.3. *For all $k \ge 4$, we have $q_1^k = 6k^3 - 2k^2$.*

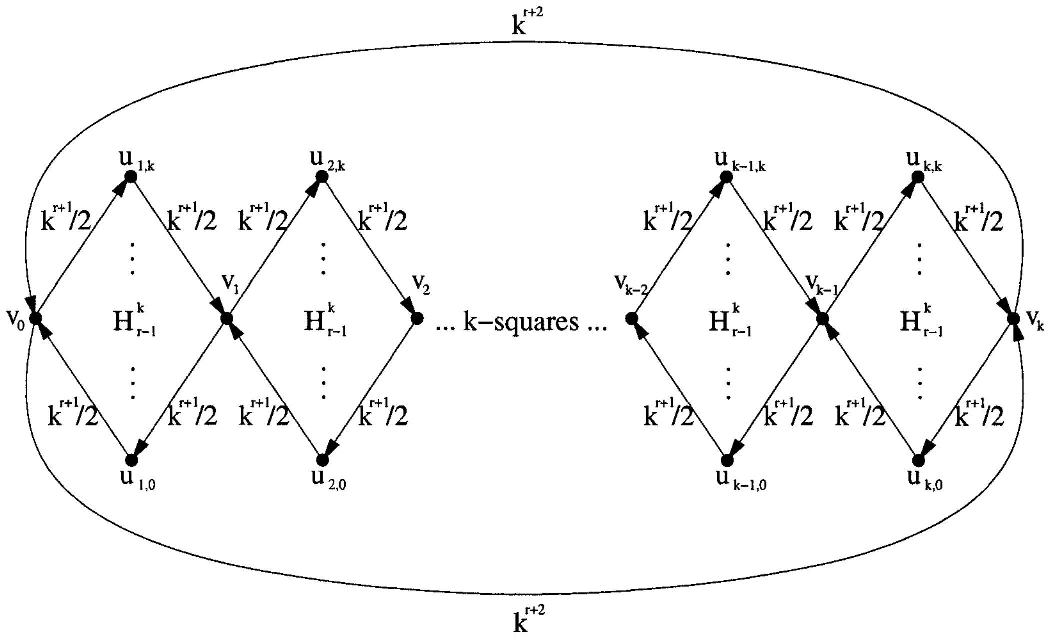

Figure 5.9: The digraph G_r^k

Proof. Let $\mathcal{C}$ be a cheapest strongly connected collection of dicycles of G_1^k which contains a dicycle of type A. Suppose, without loss of generality, that $\mathcal{C}$ contains the dicycle $(v_0, u_{1,k}, v_1, u_{2,k}, v_2, \ldots, v_{k-2}, u_{k-1,k}, v_{k-1}, u_{k,k}, v_k)$. Notice that there is no need to have the other dicycle of type A, since we could replace it with all k squares, which also have a total cost of $2k^3$, with the added benefit of covering many more nodes of G_1^k. There is no need to have the digon (v_0, v_k) in $\mathcal{C}$, since the nodes v_0 and v_k are already covered by our dicycle of type A. Lastly, there is no need to have any square or dicycle of type C, since we could replace it with all k squares in the appropriate copy of H_0^k, which also has cost $2k^2$, and covers more nodes that the square or dicycle of type C of G_1^k. Furthermore, every square in each copy of H_0^k must be present in $\mathcal{C}$ in order for our collection to be strongly connected. The only nodes that are not currently covered are the internal nodes of each k-chain within each copy of H_0^k. We can choose $k-1$ links from each k-chain to cover these nodes. Therefore, $\mathcal{C}$ contains precisely a dicycle of type A of G_1^k, all k squares in each copy of H_0^k and $k-1$ links from each of the k^2 k-chains. The total cost of this collection is hence $2k^3 + k^2(2k) + k^2(2k-2) = 6k^3 - 2k^2$. □

Lemma 5.4.4. *For all $r \geq 2$ and all $k \geq 4$, we have $q_r^k = 2k^{r+2} + k^2 m_{r-2}^k$.*

Proof. Parallelling the proof of Theorem 5.4.3, let $\mathcal{C}$ be a cheapest strongly connected collection of dicycles of G_r^k which contains a dicycle of type A. Notice that there is no need to have both the dicycles of type A, since we could replace one of them with all k squares, which also have a total cost of $2k^{r+2}$, with the added benefit of covering many more nodes of G_r^k. There is no need to have the digon (v_0, v_k) in $\mathcal{C}$, since the nodes v_0 and v_k are already covered by our dicycle of type A. Lastly, there is no need to have any square or dicycle of type C, since we could replace it with all k squares in the appropriate copy of H_{r-1}^k, which also has cost $2k^{r+1}$, and covers more nodes than the square or dicycle of type C of G_r^k. Furthermore, every square in each copy of H_{r-1}^k must be present in $\mathcal{C}$ in order for our collection to be strongly connected. Now, since $r \geq 2$, each square of each copy of H_{r-1}^k shares a pair of nodes with a unique copy of H_{r-2}^k. We simply need to find a minimum cost strongly connected collection of dicycles of H_{r-2}^k along with its associated square in H_{r-1}^k given that it must contain the square in H_{r-1}^k. However, this cost is precisely m_{r-2}^k. Now, there are exactly k squares in each of the k copies of H_{r-1}^k, so the dicycles contained within the copies of H_{r-1}^k have a total cost of $k^2 m_{r-2}^k$. These dicycles, along with the dicycle of type A of G_r^k show that $\mathcal{C}$ has total cost $2k^{r+2} + m_{r-2}^k$. □

Lemma 5.4.5. *For all $r \geq 1$ and all $k \geq 4$, we have $m_r^k = 2k^{r+2} + (k-1)m_{r-1}^k + q_{r-1}^k$.*

Proof. Let $\mathcal{C}$ be a cheapest strongly connected collection of dicycles of G_r^k which contains the digon (v_0, v_k). Similarly to the proof of Theorem 5.4.2, there is no need to have any dicycle of type A in G_r^k. Furthermore, we either have all k squares of G_r^k or we have $k-1$ squares of G_r^k along with one dicycle of type C. Consider a square of G_r^k. This square shares nodes with a unique copy of H_{r-1}^k. A minimum cost strongly connected collection of dicycles of the square along with the copy of H_{r-1}^k

which must contain the square has cost precisely m^k_{r-1}. Likewise, a minimum cost strongly connected collection of dicycles of the square along with the copy of H^k_{r-1} which must contain an intersecting dicycle of type C in G^k_r has cost precisely q^k_{r-1}. Thus

$$m^k_r = 2k^{r+2} + (k-1)m^k_{r-1} + \min\{m^k_{r-1}, q^k_{r-1}\}.$$

Now, consider a collection of dicycles which attains m^k_r for some $s \geq 0$. If we replace the digon (v_0, v_k) in this collection with a dicycle of type A in G^k_r then we get a new collection of dicycles of the same cost which is strongly connected and contains a dicycle of type A. Hence $q^k_s \leq m^k_s$. Therefore,

$$m^k_r = 2k^{r+2} + (k-1)m^k_{r-1} + q^k_{r-1}$$

as required. □

Lemma 5.4.6. *For all* $r \geq 0$ *and* $k \geq 4$, *we have* $q^k_r - k^{r+2} \leq m^k_r - 2k^{r+2}$.

Proof. We have $q^k_0 = 4k^2$ and $m^k_0 = 6k^2 - 4k + 2$. Thus

$$\begin{aligned} m^k_0 - q^k_0 - k^2 &= k^2 - 4k + 2 \\ &= k(k-4) + 2 \\ &\geq 0, \qquad \text{for } k \geq 4. \end{aligned}$$

We also have $q^k_1 = 6k^3 - 2k^2$ and $m^k_1 = 8k^3 - 6k^2 + 6k - 2$. Thus

$$\begin{aligned} m^k_1 - q^k_1 - k^3 &= k^3 - 4k^2 + 6k - 2 \\ &= k^2(k-4) + (6k-2) \\ &\geq 0, \qquad \text{for } k \geq 4. \end{aligned}$$

Lastly, we have $q^k_2 = 8k^4 - 4k^3 + 2k^2$ and $m^k_2 = 10k^4 - 8k^3 + 10k^2 - 8k + 2$. Thus

$$\begin{aligned} m^k_2 - q^k_2 - k^4 &= k^4 - 4k^3 + 8k^2 - 8k + 2 \\ &= k^3(k-4) + 8k(k-1) + 2 \\ &\geq 0, \qquad \text{for } k \geq 4. \end{aligned}$$

Thus, we see that the result holds for $r = 0$, $r = 1$, and $r = 2$. Assume that there exists some $r \geq 3$ such that the result holds for all $0 \leq s \leq r-1$. Then

$$\begin{aligned}
q_r^k - k_{r+2} &= k^{r+2} + k^2 m_{r-2}^k \text{ (by Lemma 5.4.4)} \\
&= k^{r+2} + k^2(2k^r + (k-1)m_{r-3}^k + q_{r-3}^k) \text{ (by Lemma 5.4.5)} \\
&\leq 3k^{r+2} + k^2((k-1)m_{r-3}^k + m_{r-3}^k - k^{r-1}) \text{ (by the inductive hypothesis)} \\
&= 3k^{r+2} - k^{r+1} + k^3 m_{r-3}^k \\
&= k^{r+2} - k^{r+1} + k(2k^{r+1} + k^2 m_{r-3}^k) \\
&= k^{r+2} - k^{r+1} + kq_{r-1}^k \text{ (by Lemma 5.4.4)} \\
&= (k-1)(q_{r-1} - k^{r+1}) + q_{r-1}^k + 2k^{r+2} - 2k^{r+1} \\
&\leq (k-1)(m_{r-1}^k - 2k^{r+1}) + q_{r-1}^k + 2k^{r+2} - 2k^{r+1} \text{ (by the inductive} \\
&\qquad \text{hypothesis)} \\
&= 2k^{r+2} + (k-1)m_{r-1}^k + q_{r-1}^k - 2k^{r+2} \\
&= m_r^k - 2k^{r+2} \text{ (by Lemma 5.4.5)}
\end{aligned}$$

Therefore, the result holds by strong induction on r. □

Let $\tilde{H}_r^k$ denote the digraph obtained by identifying the nodes v_0 and v_k in H_r^k. Let c_r^k be the arc costs of $\tilde{H}_r^k$ inherited from H_r^k and let x_r^k be the point of the ASEP obtained by assigning a value of $\frac{1}{2}$ to each arc of $\tilde{H}_r^k$. Let a_r^k denote the cost of a minimum-cost strongly connected collection of dicycles of $\tilde{H}_r^k$. Notice that $\tilde{H}_r^k$ has the same dicycles as G_r^k except that the digon (v_0, v_k) is not present and the two dicycles of type A have their corresponding arc v_0v_k or v_kv_0 contracted. We call these resulting dicycles type A for $\tilde{H}_r^k$.

Proposition 5.4.7. *The digraph $\tilde{H}_r^k$ has $\frac{k^{r+3}+k^{r+2}-2k}{k-1}$ nodes.*

Proof. Let n_r^k be the number of nodes of $\tilde{H}_r^k$. It is easy to compute that $n_0^k = k(k+2)$.

Furthermore,

$$\begin{aligned}\frac{k^{0+3}+k^{0+2}-2k}{k-1} &= \frac{k(k+2)(k-1)}{k-1} \\ &= k(k+2)\end{aligned}$$

as required. Also, due to the recursive definition of $\tilde{H}_r^k$ we see that $n_r^k = k(n_{r-1}^k + 2)$. If we assume that the result holds for some $r \geq 0$ then

$$\begin{aligned}n_{r+1}^k &= k\left(\frac{k^{r+3}+k^{r+2}-2k}{k-1}+2\right) \\ &= k\left(\frac{k^{r+3}+k^{r+2}-2k+2k-2}{k-1}\right) \\ &= \frac{k^{r+4}+k^{r+3}-2k}{k-1}\end{aligned}$$

and the result follows by induction on r. □

Theorem 5.4.8. *For all $r \geq 0$ and $k \geq 4$, we have $a_r^k = q_r^k - k^{r+2}$.*

Proof. Let $\mathcal{C}$ be a minimum-cost strongly connected collection of dicycles of $\tilde{H}_r^k$. Notice, that there is no need to have both dicycles of type A, since we could replace these with all k squares without affecting the cost or the connectivity.

If $\mathcal{C}$ has a single dicycle of type A, then $\mathcal{C}$ can be obtained by taking a strongly connected collection of dicycles of G_r^k which contains a single dicycle of type A and contracting the arcs v_0v_k or v_kv_0. Thus, $a_r^k + k^{r+2} \geq q_r^k$. Conversely, if we take a collection of dicycles which attains the cost q_r^k in G_r^k and contract the arc v_0v_k or v_kv_0 in the dicycle of type A, then we are left with a strongly connected collection of dicycles of $\tilde{H}_r^k$. Hence $a_r^k \leq q_r^k - k^{r+2}$ and so $a_r^k = q_r^k - k^{r+2}$.

If $\mathcal{C}$ contains no dicycle of type A, then $\mathcal{C}$ along with the digon (v_0, v_k) is a strongly connected collection of dicycles of G_r^k and so $a_r^k + 2k^{r+2} \geq m_r^k$. If we take a collection of dicycles which attains m_r^k in G_r^k and remove the digon (v_0, v_k) then we are left with a strongly connected collection of dicycles of $\tilde{H}_r^k$. Thus $a_r^k \leq m_r^k - 2k^{r+2}$ and so $a_r^k = m_r^k - 2k^{r+2}$.

So, we see that

$$a_r^k = \min\{q_r^k - k^{r+2}, m_r^k - 2k^{r+2}\}$$

and hence, by Lemma 5.4.6, $a_r^k = q_r^k - k^{r+2}$, for all $r \geq 0$ and $k \geq 4$. $\square$

Our next task is to find an explicit formula for a_r^k in terms of r and k so that we can precisely compute the integrality gap of c_r^k. That is the purpose of the following definition of ϵ_r^k.

We define

$$\epsilon_r^k = \begin{cases} 6k^2 & \text{if } r = 0 \\ 2(-1)^{\frac{r-1}{2}} k^{\frac{r+5}{2}} - 8k^2 + \sum_{j=3}^{\frac{r+3}{2}} 2(-1)^{j+1} f_r^j k^j & \text{if } r \geq 1 \text{ is odd} \\ (-1)^{\frac{r}{2}}(r+6)k^{\frac{r+4}{2}} + 8k^2 + \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^j f_r^j k^j & \text{if } r \geq 2 \text{ is even} \end{cases}$$

where

$$f_r^j = \binom{r-j+2}{j-3} + \binom{r-j+1}{j-2} + 3\binom{r-j+2}{j-2}.$$

Lemma 5.4.9. *For every $r \geq 2$ and every $j \geq 4$ we have $f_r^j = f_{r-1}^j + f_{r-2}^{j-1}$.*

Proof.

$$\begin{aligned} f_{r-1}^j + f_{r-2}^{j-1} &= \binom{r-j+1}{j-3} + \binom{r-j}{j-2} + 3\binom{r-j+1}{j-2} \\ &\quad + \binom{r-j+1}{j-4} + \binom{r-j}{j-3} + 3\binom{r-j+1}{j-3} \\ &= \binom{r-j+1}{j-3} + \binom{r-j+1}{j-4} + \binom{r-j}{j-2} + \binom{r-j}{j-3} \\ &\quad + 3\binom{r-j+1}{j-2} + 3\binom{r-j+1}{j-3} \\ &= \binom{r-j+2}{j-3} + \binom{r-j+1}{j-2} + 3\binom{r-j+2}{j-2} \\ &= f_r^j \end{aligned}$$

$\square$

Lemma 5.4.10. *For all $k \geq 4$ and $r \geq 3$, we have $\epsilon_r^k = (k-1)\epsilon_{r-1}^k + k^2\epsilon_{r-3}^k$.*

Proof. We have

$$\begin{aligned}(k-1)\epsilon_2^k + k^2\epsilon_0^k &= (k-1)(-8k^3+8k^2)+k^2(6k^2)\\ &= -8k^4+8k^3+8k^3-8k^2+6k^4\\ &= -2k^4+16k^3-8k^2\\ &= \epsilon_3^k.\end{aligned}$$

So the result holds for $r=3$.

Now let $r \geq 4$ be even. Then

$$\begin{aligned}&(k-1)\epsilon_{r-1}^k + k^2\epsilon_{r-3}^k\\ &= (k-1)(2(-1)^{\frac{r-2}{2}}k^{\frac{r+4}{2}} - 8k^2 + \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^{j+1} f_{r-1}^j k^j)\\ &\quad +k^2(2(-1)^{\frac{r-4}{2}}k^{\frac{r+2}{2}} - 8k^2 + \sum_{j=3}^{\frac{r}{2}} 2(-1)^{j+1} f_{r-3}^j k^j)\\ &= 2(-1)^{\frac{r-2}{2}}k^{\frac{r+6}{2}} - 8k^3 + 2(-1)^{\frac{r}{2}}k^{\frac{r+4}{2}} + 8k^2 + 2(-1)^{\frac{r-4}{2}}k^{\frac{r+6}{2}}\\ &\quad -8k^4 + \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^{j+1} f_{r-1}^j k^{j+1} - \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^{j+1} f_{r-1}^j k^j\\ &\quad + \sum_{j=3}^{\frac{r}{2}} 2(-1)^{j+1} f_{r-3}^j k^{j+2}\\ &= 2(-1)^{\frac{r}{2}}k^{\frac{r+4}{2}} - 8k^4 - 8k^3 + 8k^2 + \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^{j+1} f_{r-1}^j k^{j+1}\\ &\quad - \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^{j+1} f_{r-1}^j k^j + \sum_{j=3}^{\frac{r}{2}} 2(-1)^{j+1} f_{r-3}^j k^{j+2}\\ &= 2(-1)^{\frac{r}{2}}k^{\frac{r+4}{2}} - 8k^4 - 8k^3 + 8k^2 + \sum_{j=4}^{\frac{r+4}{2}} 2(-1)^{j} f_{r-1}^{j-1} k^j\\ &\quad - \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^{j+1} f_{r-1}^j k^j + \sum_{j=5}^{\frac{r+4}{2}} 2(-1)^{j-1} f_{r-3}^{j-2} k^j\end{aligned}$$

$$
\begin{aligned}
&= 2(-1)^{\frac{r}{2}}k^{\frac{r+4}{2}} - 8k^4 - 8k^3 + 8k^2 + \sum_{j=4}^{\frac{r+4}{2}} 2(-1)^j f_{r-1}^{j-1}k^j \\
&\quad + \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^j f_{r-1}^{j}k^j - \sum_{j=5}^{\frac{r+4}{2}} 2(-1)^j f_{r-3}^{j-2}k^j \\
&= 2(-1)^{\frac{r}{2}}k^{\frac{r+4}{2}} - 8k^4 - 8k^3 + 8k^2 + 2f_{r-1}^3k^4 + 2(-1)^{\frac{r+4}{2}} f_{r-1}^{\frac{r+2}{2}} k^{\frac{r+4}{2}} \\
&\quad -2f_{r-1}^3k^3 + 2f_{r-1}^4k^4 - 2(-1)^{\frac{r+4}{2}} f_{r-3}^{\frac{r}{2}}k^{\frac{r+4}{2}} + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j f_{r-1}^{j-1}k^j \\
&\quad + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j f_{r-1}^{j}k^j - \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j f_{r-3}^{j-2}k^j \\
&= (-1)^{\frac{r}{2}}(2 + 2f_{r-1}^{\frac{r+2}{2}} - 2f_{r-3}^{\frac{r}{2}})k^{\frac{r+4}{2}} + 8k^2 + 2(-1)^3(f_{r-1}^3 + 4)k^3 \\
&\quad +2(-1)^4(f_{r-1}^3 + f_{r-1}^4 - 4)k^4 + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j (f_{r-1}^{j-1} + f_{r-1}^{j} - f_{r-3}^{j-2})k^j \\
&= (-1)^{\frac{r}{2}}(r+6)k^{\frac{r+4}{2}} + 8k^2 + 2(-1)^3(4r-4)k^3 \\
&\quad +2(-1)^4(2r^2 - 10r + 13)k^4 + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j (f_{r-1}^{j-1} + f_{r-1}^{j} - f_{r-3}^{j-2})k^j \\
&= (-1)^{\frac{r}{2}}(r+6)k^{\frac{r+4}{2}} + 8k^2 + 2(-1)^3 f_r^3k^3 + 2(-1)^4 f_r^4k^4 \\
&\quad + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j (f_{r-1}^{j-1} + f_{r-1}^{j} - f_{r-3}^{j-2})k^j \\
&= (-1)^{\frac{r}{2}}(r+6)k^{\frac{r+4}{2}} + 8k^2 + 2(-1)^3 f_r^3k^3 + 2(-1)^4 f_r^4k^4 \\
&\quad + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j ((f_{r-2}^{j-1} + f_{r-3}^{j-2}) + f_{r-1}^{j} - f_{r-3}^{j-2})k^j \\
&= (-1)^{\frac{r}{2}}(r+6)k^{\frac{r+4}{2}} + 8k^2 + 2(-1)^3 f_r^3k^3 + 2(-1)^4 f_r^4k^4 \\
&\quad + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j (f_{r-2}^{j-1} + f_{r-1}^{j})k^j \\
&= (-1)^{\frac{r}{2}}(r+6)k^{\frac{r+4}{2}} + 8k^2 + 2(-1)^3 f_r^3k^3 + 2(-1)^4 f_r^4k^4 + \sum_{j=5}^{\frac{r+2}{2}} 2(-1)^j f_r^jk^j
\end{aligned}
$$

$$
\begin{aligned}
&= (-1)^{\frac{r}{2}}(r+6)k^{\frac{r+4}{2}} + 8k^2 + \sum_{j=3}^{\frac{r+2}{2}} 2(-1)^j f_r^j k^j \\
&= \epsilon_r^k.
\end{aligned}
$$

Now let $r \geq 5$ be odd. Then

$$
\begin{aligned}
&(k-1)\epsilon_{r-1}^k + k^2\epsilon_{r-3}^k \\
&= (k-1)((-1)^{\frac{r-1}{2}}(r+5)k^{\frac{r+3}{2}} + 8k^2 + \sum_{j=3}^{\frac{r+1}{2}} 2(-1)^j f_{r-1}^j k^j) \\
&\quad +k^2((-1)^{\frac{r-3}{2}}(r+3)k^{\frac{r+1}{2}} + 8k^2 + \sum_{j=3}^{\frac{r-1}{2}} 2(-1)^j f_{r-3}^j k^j) \\
&= (-1)^{\frac{r-1}{2}}(r+5)k^{\frac{r+5}{2}} + 8k^3 + (-1)^{\frac{r+1}{2}}(r+5)k^{\frac{r+3}{2}} - 8k^2 \\
&\quad +(-1)^{\frac{r-3}{2}}(r+3)k^{\frac{r+5}{2}} + 8k^4 + \sum_{j=3}^{\frac{r+1}{2}} 2(-1)^j f_{r-1}^j k^{j+1} \\
&\quad -\sum_{j=3}^{\frac{r+1}{2}} 2(-1)^j f_{r-1}^j k^j + \sum_{j=3}^{\frac{r-1}{2}} 2(-1)^j f_{r-3}^j k^{j+2} \\
&= 2(-1)^{\frac{r-1}{2}}k^{\frac{r+5}{2}} + (-1)^{\frac{r+1}{2}}(r+5)k^{\frac{r+3}{2}} + 8k^4 + 8k^3 - 8k^2 \\
&\quad +\sum_{j=4}^{\frac{r+3}{2}} 2(-1)^{j-1} f_{r-1}^{j-1} k^j - \sum_{j=3}^{\frac{r+1}{2}} 2(-1)^j f_{r-1}^j k^j + \sum_{j=5}^{\frac{r+3}{2}} 2(-1)^{j-2} f_{r-3}^{j-2} k^j \\
&= 2(-1)^{\frac{r-1}{2}}k^{\frac{r+5}{2}} + (-1)^{\frac{r+5}{2}}(r+5)k^{\frac{r+3}{2}} + 8k^4 + 8k^3 - 8k^2 \\
&\quad +\sum_{j=4}^{\frac{r+3}{2}} 2(-1)^{j+1} f_{r-1}^{j-1} k^j + \sum_{j=3}^{\frac{r+1}{2}} 2(-1)^{j+1} f_{r-1}^j k^j - \sum_{j=5}^{\frac{r+3}{2}} 2(-1)^{j+1} f_{r-3}^{j-2} k^j \\
&= 2(-1)^{\frac{r-1}{2}}k^{\frac{r+5}{2}} + 2(-1)^{\frac{r+5}{2}}(\frac{r+5}{2} + 2f_{r-1}^{\frac{r+1}{2}} - 2f_{r-3}^{\frac{r-1}{2}})k^{\frac{r+3}{2}} \\
&\quad +2(-1)^5(f_{r-1}^3 + f_{r-1}^4 - 4)k^4 + 2(-1)^4(f_{r-1}^3 + 4)k^3 - 8k^2 \\
&\quad +\sum_{j=5}^{\frac{r+1}{2}} 2(-1)^{j+1} f_{r-1}^{j-1} k^j + \sum_{j=5}^{\frac{r+1}{2}} 2(-1)^{j+1} f_{r-1}^j k^j - \sum_{j=5}^{\frac{r+1}{2}} 2(-1)^{j+1} f_{r-3}^{j-2} k^j
\end{aligned}
$$

$$\begin{aligned}
&= 2(-1)^{\frac{r-1}{2}}k^{\frac{r+5}{2}} + 2(-1)^{\frac{r+5}{2}}(\frac{1}{8}r^2 + \frac{3}{2}r + \frac{19}{8})k^{\frac{r+3}{2}} \\
&\quad +2(-1)^5(2r^2 - 10r + 13)k^4 + 2(-1)^4(4r-4)k^3 - 8k^2 \\
&\quad + \sum_{j=5}^{\frac{r+1}{2}} 2(-1)^{j+1}(f_{r-1}^{j-1} + f_{r-1}^{j} - f_{r-3}^{j-2})k^j \\
&= 2(-1)^{\frac{r-1}{2}}k^{\frac{r+5}{2}} + 2(-1)^{\frac{r+5}{2}} f_r^{\frac{r+3}{2}} k^{\frac{r+3}{2}} + 2(-1)^5 f_r^4 k^4 + 2(-1)^4 f_r^3 k^3 \\
&\quad -8k^2 + \sum_{j=5}^{\frac{r+1}{2}} 2(-1)^{j+1}((f_{r-2}^{j-1} + f_{r-3}^{j-2}) + f_{r-1}^{j} - f_{r-3}^{j-2})k^j \\
&= 2(-1)^{\frac{r-1}{2}}k^{\frac{r+5}{2}} + 2(-1)^{\frac{r+5}{2}} f_r^{\frac{r+3}{2}} k^{\frac{r+3}{2}} + 2(-1)^5 f_r^4 k^4 + 2(-1)^4 f_r^3 k^3 \\
&\quad -8k^2 + \sum_{j=5}^{\frac{r+1}{2}} 2(-1)^{j+1} f_r^j k^j \\
&= 2(-1)^{\frac{r-1}{2}}k^{\frac{r+5}{2}} - 8k^2 + \sum_{j=3}^{\frac{r+3}{2}} 2(-1)^{j+1} f_r^j k^j \\
&= \epsilon_r^j.
\end{aligned}$$

□

Lemma 5.4.11. *For every $r \geq 0$ and $k \geq 4$, we have $\epsilon_r^k \geq -\frac{3}{2}k^{r+2}$.*

Proof. If $k \geq 4$ then $\epsilon_0^k = 4k^2 \geq -\frac{3}{2}k^2$. Now,

$$\begin{aligned}
\epsilon_1^k &= 2k^3 - 8k^2 \\
&= 2(k-4)k^2 \\
&\geq 0 \text{ for } k \geq 4 \\
&\geq -\frac{3}{2}k^3
\end{aligned}$$

and

$$\begin{aligned}
\epsilon_2^k &= -8k^3 + 8k^2 \\
&= -\frac{3}{2}k^4 + \frac{3}{2}k^4 - 8k^3 + 8k^2 \\
&= -\frac{3}{2}k^4 + k^2(k-4)(\frac{3}{2}k - 2) \\
&\geq -\frac{3}{2}k^4, \text{ since } k \geq 4.
\end{aligned}$$

Thus for the base cases $r = 0$, $r = 1$, and $r = 2$ the result holds.

We now proceed by strong induction on r using the recursive relationship

$$\epsilon_r^k = (k-1)\epsilon_{r-1}^k + k^2\epsilon_{r-3}^k.$$

Assume that for some $r \geq 3$ we have that

$$\epsilon_s^k \geq -\frac{3}{2}k^{s+2}$$

for every $0 \leq s \leq r-1$. Then

$$\begin{aligned} \epsilon_r^k &= (k-1)\epsilon_{r-1}^k + k^2\epsilon_{r-3}^k \\ &\geq (k-1)(-\frac{3}{2}k^{r+1}) + k^2(-\frac{3}{2}k^{r-1}) \\ &= -\frac{3}{2}k^{r+2} + \frac{3}{2}k^{r+1} - \frac{3}{2}k^{r+1} \\ &= -\frac{3}{2}k^{r+2} \end{aligned}$$

and the result follows by strong induction. □

Theorem 5.4.12. *For all $k \geq 4$ and all $r \geq 0$, we have*

$$q_r^k = \frac{(2r+4)k^{r+4} + (6r+16)k^{r+3} + (4r+10)k^{r+2} + \epsilon_r^k}{k^2+4k+4}$$

and

$$m_r^k = \frac{(2r+6)k^{r+4} + (6r+20)k^{r+3} + (4r+10)k^{r+2} + \epsilon_{r+2}^k/k^2}{k^2+4k+4}.$$

Proof. If $r = 0$ then

$$\begin{aligned} &\frac{(2(0)+4)k^{0+4} + (6(0)+16)k^{0+3} + (4(0)+10)k^{0+2} + \epsilon_0^k}{k^2+4k+4} \\ &= \frac{4k^4 + 16k^3 + 10k^2 + 6k^2}{k^2+4k+4} \\ &= \frac{4k^4 + 16k^3 + 16k^2}{k^2+4k+4} \\ &= 4k^2 \\ &= q_0^k \end{aligned}$$

and

$$\begin{aligned}
&\frac{(2(0)+6)k^{0+4}+(6(0)+20)k^{0+3}+(4(0)+10)k^{0+2}+\epsilon^k_{0+2}/k^2}{k^2+4k+4}\\
&= \frac{6k^4+20k^3+10k^2+(-8k^3+8k^2)/k^2}{k^2+4k+4}\\
&= \frac{6k^4+20k^3+10k^2-8k+8}{k^2+4k+4}\\
&= 6k^2-4k+2\\
&= m^k_0.
\end{aligned}$$

If $r = 1$ then

$$\begin{aligned}
&\frac{(2(1)+4)k^{1+4}+(6(1)+16)k^{1+3}+(4(1)+10)k^{1+2}+\epsilon^k_1}{k^2+4k+4}\\
&= \frac{6k^5+22k^4+14k^3+(2k^3-8k^2)}{k^2+4k+4}\\
&= \frac{6k^5+22k^4+16k^3-8k^2}{k^2+4k+4}\\
&= 6k^3-2k^2\\
&= q^k_1
\end{aligned}$$

and

$$\begin{aligned}
&\frac{(2(1)+6)k^{1+4}+(6(1)+20)k^{1+3}+(4(1)+10)k^{1+2}+\epsilon^k_{1+2}/k^2}{k^2+4k+4}\\
&= \frac{8k^5+26k^4+14k^3+(-2k^4+16k^3-8k^2)/k^2}{k^2+4k+4}\\
&= \frac{8k^5+26k^4+14k^3-2k^2+16k-8}{k^2+4k+4}\\
&= 8k^3-6k^2+6k-2\\
&= m^k_1.
\end{aligned}$$

Hence the base cases hold for $r = 0$ and $r = 1$.

We proceed by strong induction on r using the recursive relationships

$$q^k_r = 2k^{r+2} + k^2 m^k_{r-2}$$

and

$$m_r^k = 2k^{r+2} + (k-1)m_{r-1}^k + q_{r-1}^k.$$

Assume that there is some $r \geq 2$ such that for every $0 \leq s \leq r-1$ we have that

$$q_s^k = \frac{(2s+4)k^{s+4} + (6s+16)k^{s+3} + (4s+10)k^{s+2} + \epsilon_s^k}{k^2+4k+4}$$

and

$$m_s^k = \frac{(2s+6)k^{s+4} + (6s+20)k^{s+3} + (4s+10)k^{s+2} + \epsilon_{s+2}^k/k^2}{k^2+4k+4}.$$

$$\begin{aligned}
q_r^k &= 2k^{r+2} + k^2 m_{r-2}^k \\
&= 2k^{r+2} + k^2\frac{(2r+2)k^{r+2} + (6r+8)k^{r+1} + (4r+2)k^r + \epsilon_r^k/k^2}{k^2+4k+4} \\
&= 2k^{r+2} + \frac{(2r+2)k^{r+4} + (6r+8)k^{r+3} + (4r+2)k^{r+2} + \epsilon_r^k}{k^2+4k+4} \\
&= \frac{2k^{r+4} + 8k^{r+3} + 8k^{r+2} + (2r+2)k^{r+4} + (6r+8)k^{r+3} + (4r+2)k^{r+2} + \epsilon_r^k}{k^2+4k+4} \\
&= \frac{(2r+4)k^{r+4} + (6r+16)k^{r+3} + (4r+10)k^{r+2} + \epsilon_r^k}{k^2+4k+4}
\end{aligned}$$

$$\begin{aligned}
m_r^k &= 2k^{r+2} + (k-1)m_{r-1}^k + q_{r-1}^k \\
&= 2k^{r+2} + (k-1)\frac{(2r+4)k^{r+3} + (6r+14)k^{r+2} + (4r+6)k^{r+1} + \epsilon_{r+1}^k/k^2}{k^2+4k+4} \\
&\quad + \frac{(2r+2)k^{r+3} + (6r+10)k^{r+2} + (4r+6)k^{r+1} + \epsilon_{r-1}^k}{k^2+4k+4} \\
&= 2k^{r+2} + \frac{(2r+2)k^{r+3} + (6r+10)k^{r+2} + (4r+6)k^{r+1} + \epsilon_{r-1}^k}{k^2+4k+4} \\
&\quad + \frac{(2r+4)k^{r+4} + (4r+10)k^{r+3} - (2r+8)k^{r+2} - (4r+6)k^{r+1} + (k-1)\epsilon_{r+1}^k/k^2}{k^2+4k+4} \\
&= \frac{(2r+6)k^{r+4} + (6r+20)k^{r+3} + (4r+10)k^{r+2} + ((k-1)\epsilon_{r+1}^k + k^2\epsilon_{r-1}^k)/k^2}{k^2+4k+4} \\
&= \frac{(2r+6)k^{r+4} + (6r+20)k^{r+3} + (4r+10)k^{r+2} + \epsilon_{r+2}^k/k^2}{k^2+4k+4} \text{ (by Lemma 5.4.10).}
\end{aligned}$$

Therefore, the result holds for all $r \geq 0$ by strong induction. □

Corollary 5.4.13. *For all $r \geq 0$ and all $k \geq 4$, we have*

$$a_r^k = \frac{(2r+3)k^{r+4} + (6r+12)k^{r+3} + (4r+6)k^{r+2} + \epsilon_r^k}{k^2+4k+4}.$$

Proof. Since $a_r^k = k^{r+2} + k^2 m_{r-2}^k$ and $q_r^k = 2k^{r+2} + k^2 m_{r-2}^k$, we have

$$\begin{aligned} a_r^k &= q_r^k - k^{r+2} \\ &= \frac{(2r+4)k^{r+4} + (6r+16)k^{r+3} + (4r+10)k^{r+2} + \epsilon_r^k}{k^2+4k+4} - k^{r+2} \\ &= \frac{(2r+3)k^{r+4} + (6r+12)k^{r+3} + (4r+6)k^{r+2} + \epsilon_r^k}{k^2+4k+4}. \end{aligned}$$

□

Corollary 5.4.14. *For all $r \geq 0$ and all $k \geq 4$, we have*

$$a_r^k \geq \frac{(2r+3)k^{r+4} + (6r+12)k^{r+3} + (4r+9/2)k^{r+2}}{k^2+4k+4}.$$

Proof. From Lemma 5.4.11, we know that $\epsilon_r^k \geq -\frac{3}{2}k^{r+2}$ and so

$$\begin{aligned} a_r^k &\geq \frac{(2r+3)k^{r+4} + (6r+12)k^{r+3} + (4r+6)k^{r+2} - \frac{3}{2}k^{r+2}}{k^2+4k+4} \\ &= \frac{(2r+3)k^{r+4} + (6r+12)k^{r+3} + (4r+9/2)k^{r+2}}{k^2+4k+4}. \end{aligned}$$

□

Theorem 5.4.15. *Extreme point x_r^k is an optimal solution the $ATSP_{LP}$ with respect to c_r^k for each $r \geq 0$ and $k \geq 4$.*

Proof. We define y-values to the nodes of $D(x_r^k)$ and use the complementary slackness conditions to prove optimality. Consider any node v of $D(x_r^k)$. If v is incident to an arc of cost 1 then assign

$$y_v^{\text{in}} = y_v^{\text{out}} = \frac{1}{2}.$$

Otherwise, if the cheapest arc incident to v has cost $k^s/2$ then assign

$$y_v^{\text{in}} = y_v^{\text{out}} = \frac{k^{s+1} + (-1)^s}{2k+2}.$$

Let $uv \in E(x_r^k)$. If $(c_r^k)_{uv} = 1$ then

$$\begin{aligned} y_u^{\text{out}} + y_v^{\text{in}} &= \frac{1}{2} + \frac{1}{2} \\ &= 1. \end{aligned}$$

If $(c_r^k)_{uv} = k/2$ for some arc uv, then one endpoint of uv is incident to a arc of cost 1 and the cheapest arc incident to the other endpoint has cost $k/2$. Hence,

$$\begin{aligned} y_u^{\text{out}} + y_v^{\text{in}} &= \frac{1}{2} + \frac{k^2 + (-1)^1}{2k+2} \\ &= \frac{k + 1 + k^2 - 1}{2(k+1)} \\ &= \frac{k(k+1)}{2(k+1)} \\ &= \frac{k}{2}. \end{aligned}$$

Otherwise, if $(c_r^k)_{uv} = k^s/2$ for some $2 \leq s \leq r+1$, then the cheapest arc incident to one of the endpoints of uv has cost $k^{s-1}/2$ and the cheapest arc incident to the other endpoint has cost $k^s/2$. Thus,

$$\begin{aligned} y_u^{\text{out}} + y_v^{\text{in}} &= \frac{k^s + (-1)^{s-1}}{2k+2} + \frac{k^{s+1} + (-1)^s}{2k+2} \\ &= \frac{(k+1)k^s + (1-1)(-1)^{s-1}}{2k+2} \\ &= \frac{(k+1)k^s}{2(k+1)} \\ &= \frac{k^s}{2}. \end{aligned}$$

Therefore, for all the arcs of $E(x_r^k)$ the complementary slackness conditions hold.

Now consider some $uv \notin E(x_r^k)$. Then the cost of uv is the cost of a cheapest (u,v)-dipath in $D(x_r^k)$. If this dipath has internal nodes $w_1, \ldots, w_t$ then

$$(c_r^k)_{uv} = (y_u^{\text{out}} + y_{w_1}^{\text{in}}) + (y_{w_1}^{\text{out}} + y_{w_2}^{\text{in}}) + \ldots + (y_{w_t}^{\text{out}} + y_v^{\text{in}}).$$

But, all the y-values are positive so

$$(c_r^k)_{uv} > y_u^{\text{out}} + y_v^{\text{in}}$$

and the complementary slackness condition holds for uv. Therefore, x_r^k is optimal with respect to c_r^k. □

Lemma 5.4.16. *For all $r \geq 0$ and $k \geq 4$, we have $c_r^k x_r^k = (r+2)k^{r+2}$.*

Proof. Due to the recursive construction of x_r^k we see that

$$c_r^k x_r^k = k(k^{r+1} + c_{r-1}^k x_{r-1}^k).$$

It is straightforward to compute that

$$c_0^k x_0^k = 2k^2$$

so the result holds for $r = 0$. Assume that the result holds for some $r \geq 0$ then

$$\begin{aligned} c_{r+1}^k x_{r+1}^k &= k(k^{r+2} + c_r^k x_r^k) \\ &= k(k^{r+2} + (r+2)k^{r+2}) \\ &= (r+3)k^{r+3}. \end{aligned}$$

Hence the result follows by induction. □

Theorem 5.4.17. *For all $r \geq 0$ and all $k \geq 4$, we have*

$$\begin{aligned} Gap(c_r^k) &\geq \frac{(2r+3)k^2 + (6r+12)k + (4r+9/2)}{(r+2)(k^2+4k+4)} \\ &= \left(2 - \frac{1}{r+2}\right)\left(1 - \frac{1}{k+2}\right) + \frac{6k-3}{2(r+2)(k+2)^2}. \end{aligned}$$

Proof. By Corollary 5.4.13 and Lemma 5.4.16 we have

$$\begin{aligned} \mathrm{Gap}(c_r^k) &= \frac{a_r^k}{c_r^k x_r^k} \\ &\geq \frac{(2r+3)k^{r+4} + (6r+12)k^{r+3} + (4r+9/2)k^{r+2}}{(r+2)k^{r+2}(k^2+4k+4)} \\ &= \frac{(2r+3)k^2 + (6r+12)k + (4r+9/2)}{(r+2)(k^2+4k+4)} \end{aligned}$$

and

$$\begin{aligned}
&\left(2-\frac{1}{r+2}\right)\left(1-\frac{1}{k+2}\right)+\frac{6k-3}{2(r+2)(k+2)^2}\\
=&\left(\frac{2r+3}{r+2}\right)\left(\frac{k+1}{k+2}\right)+\frac{3k-3/2}{(r+2)(k+2)^2}\\
=&\frac{(2r+3)(k+1)(k+2)}{(r+2)(k+2)^2}+\frac{3k-3/2}{(r+2)(k+2)^2}\\
=&\frac{(2r+3)k^2+(6r+9)k+(4r+6)}{(r+2)(k^2+4k+4)}+\frac{3k-3/2}{(r+2)(k^2+4k+4)}\\
=&\frac{(2r+3)k^2+(6r+12)k+(4r+9/2)}{(r+2)(k^2+4k+4)}.
\end{aligned}$$

□

The important consequence of Theorem 5.4.17 is that

$$\lim_{r\to\infty} \mathrm{Gap}(c_r^k) = 2 - \frac{2}{k+2}$$

and

$$\lim_{k\to\infty} \mathrm{Gap}(c_r^k) = 2 - \frac{1}{r+2}.$$

Therefore, as r and k both approach ∞, we have that $\mathrm{Gap}(c_r^k)$ approaches 2. This refutes Carr and Vempala's [16] conjecture that the largest integrality gap possible for the metric ATSP is 4/3. However, we are not the only ones to refute this conjecture. Charikar, Goemans, and Karloff [18] independently discovered that the largest integrality gap for the metric ATSP is at least 2 at the same time as we had found our family of extreme points that proves that the integrality gap is at least 3/2. Their work motivates our use of recursion in generalizing our family of digraphs G_0^k to the recursive family of extreme points x_r^k. We now turn our attention to their important result and we compare it to our own.

5.5 Big Integrality Gaps by Charikar, Goemans, and Karloff

In their paper from 2004, Charikar, Goemans, and Karloff [18] introduced a family of half-integer extreme points of the ASEP whose integrality gap approaches 2 as the members of the family have more and more nodes. The importance of an integrality gap of 2 (as they noted in their paper) is that this is the largest possible integrality gap for a half-integer extreme point of the ASEP. This can be seen by taking any half-integer extreme point of the ASEP and replacing any 1-arc with two parallel arcs. The resulting digraph is Eulerian and its cost is at most twice that of the original half-integer extreme point.

We now describe the recursive family of extreme points presented in their paper which we call L_t^s. We start with a dicycle of length $2s+2$ and s $(s+1)$-chains as shown in Figure 5.10, which we call $\tilde{L}_0^s$. Each arc in the $(s+1)$-chains has a cost of 1 and the remaining arcs have cost s. We obtain L_0^s from $\tilde{L}_0^s$ by identifying the nodes u_0 and u_{2s+1}. To find L_t^s for some $t \geq 1$, we take L_{t-1}^s, multiply all its arc costs by s, and then replace each $(s+1)$-chain with a copy of $\tilde{L}_0^s$.

Let d_t^s be the metric completion of the arc costs of L_t^s. In their paper, Charikar, Goemans, and Karloff prove the following.

Theorem 5.5.1. *For all $t \geq 0$ and $s \geq 3$, we have*

$$Gap(d_t^s) \geq \left(2 - \frac{1}{t+2}\right)\left(1 - \frac{2}{s+1}\right).$$

They also mention in their paper that L_t^s has $\frac{s^{t+3}+s^{t+2}-s-1}{s-1}$ nodes. Now suppose that $r = t$ and $k = s$, for some $t \geq 0$ and $s \geq 4$. Then L_t^s has $\frac{s^{t+3}+s^{t+2}-s-1}{s-1}$ nodes and $\tilde{H}_t^s$ has $\frac{s^{t+3}+s^{t+2}-2s}{s-1}$ nodes by Proposition 5.4.7. Since $s \geq 4$ this means that $\tilde{H}_t^s$ has strictly fewer nodes than L_t^s. Furthermore, when we compare the lower bounds

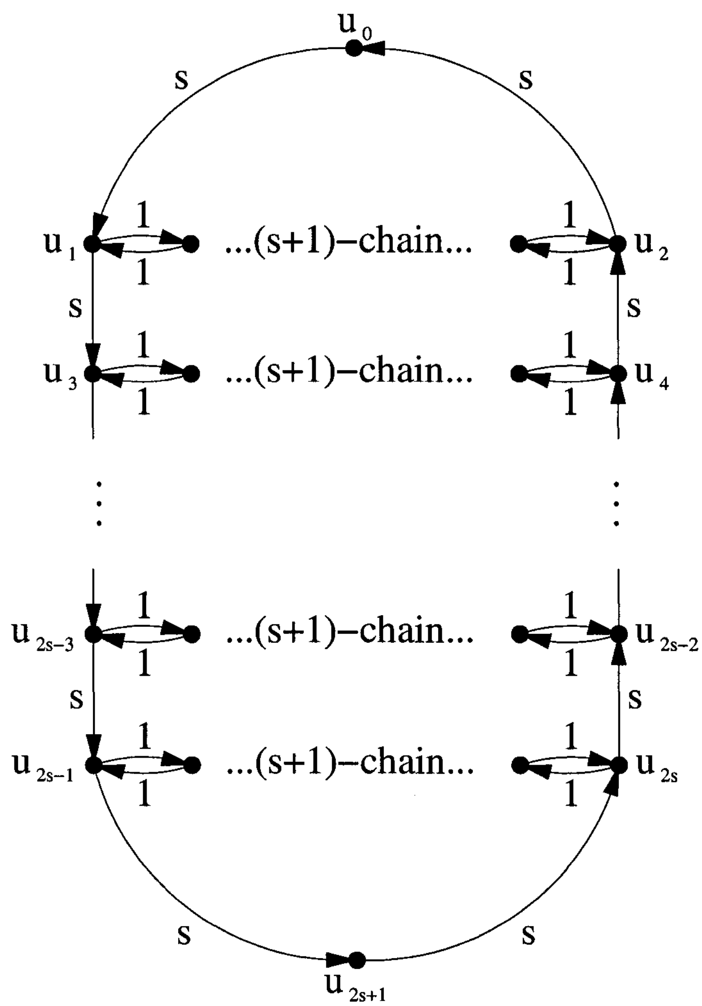

Figure 5.10: The digraph $\tilde{L}_0^s$

on the integrality gaps associated with these two digraphs, we see that

$$\begin{aligned} \mathrm{Gap}(c_t^s) &= \left(2-\frac{1}{t+2}\right)\left(1-\frac{1}{s+2}\right)+\frac{6s-3}{2(t+2)(s+2)^2} \\ &> \left(2-\frac{1}{t+2}\right)\left(1-\frac{1}{s+2}\right) \quad \text{for } s \geq 4 \\ &> \left(2-\frac{1}{t+2}\right)\left(1-\frac{2}{s+1}\right). \end{aligned}$$

Therefore, the lower bound on $\mathrm{Gap}(c_t^s)$ is strictly larger than that of $\mathrm{Gap}(d_t^s)$.

To compare the two lower bounds on the integrality gaps, we consider the ratio of how close each of the bounds are to 2 and take the limit as t approaches ∞.

$$\begin{aligned}
&\lim_{t\to\infty} \frac{2-\left(2-\frac{1}{t+2}\right)\left(1-\frac{1}{s+2}\right)-\frac{6s-3}{2(t+2)(s+2)^2}}{2-\left(2-\frac{1}{t+2}\right)\left(1-\frac{2}{s+1}\right)} \\
&= \frac{2-2\left(\frac{s+1}{s+2}\right)}{2-2\left(\frac{s-1}{s+1}\right)} \\
&= \left(\frac{(s+2)-(s+1)}{s+2}\right)\left(\frac{s+1}{(s+1)-(s-1)}\right) \\
&= \frac{s+1}{2s+4} \\
&= \frac{1}{2}-\frac{1}{2s+4}
\end{aligned}$$

Hence, for very large t, the lower bound on $\mathrm{Gap}(c_t^s)$ is twice as close to 2 as the lower bound on $\mathrm{Gap}(d_t^s)$. Furthermore, as noted above, the larger gap is accomplished on a digraph with fewer nodes.

We see in this chapter how we develop a family of half-integer extreme points with large integrality gaps based on the data we generate in the previous chapter. Our first generalization allows us to approach an integrality gap of 3/2. Our second generalization allows us to approach an integrality gap of 2. We finish by showing that our family has certain advantages over that of Charikar, Goemans, and Karloff.

Chapter 6

The Strength of ATSP Facets

In previous chapters, we investigate the integrality gap starting with the extreme points of the ASEP. In this chapter, we turn our attention to generating metric cost functions and then computing the integrality gap for each of them. However, there are infinitely many metric cost functions that we could check. How do we limit our search? We decide to narrow our focus to a set of metric cost functions which can be obtained from certain famous classes of facets of $\mathcal{P}_T^n$. Not only are these facets well-studied, but we see that the integrality gap is relatively easy to compute since the optimal value of the ATSP is given by the particular facet-inducing inequality. More importantly, these integrality gaps may tell us something about the effectiveness of adding the corresponding inequalities in the cutting-plane approach to solving the ATSP.

6.1 Facet-inducing Inequalities and the Integrality Gap

As we see in the Polyhedral Theory section of Chapter 2, two facet-inducing inequalities of $\mathcal{P}_T^n$ are equivalent (and hence induce the same facet) if one is a positive multiple of the other, plus any linear combination of the node equalities. More rigorously, two facet-inducing inequalities $\alpha x \geq \alpha_0$ and $\beta x \geq \beta_0$ of the ATSP-polytope are equivalent if there is some $\lambda > 0$, $y^{\text{in}} \in \mathbb{R}^V$, and $y^{\text{out}} \in \mathbb{R}^V$ such that

$$\beta_{uv} = \lambda\alpha_{uv} + y_u^{\text{out}} + y_v^{\text{in}} \qquad \text{for each } uv \in E$$

and

$$\beta_0 = \lambda\alpha_0 + \sum_{v \in V}(y_v^{\text{out}} + y_v^{\text{in}}).$$

Geometrically, we are tilting the hyperplane $\alpha x = \alpha_0$ to obtain the hyperplane $\beta x = \beta_0$ using the equality set of the ASEP.

Now, if we have some facet-inducing inequality $\alpha x \geq \alpha_0$ for the ATSP-polytope, we can think of each α_{uv} as a cost on the arc uv. By doing this, we can immediately conclude that ATSP$(\alpha) = \alpha_0$. In previous chapters, we had a lot of difficulty in computing the optimal value of the ATSP whereas if we use the coefficients of a facet-inducing inequality as the arc costs, then the optimal value is given. We now turn our attention to studying the relationship between the integrality gap of two equivalent facet-inducing inequalities.

Lemma 6.1.1. *Let $\alpha x \geq \alpha_0$ and $\beta x \geq \beta_0$ be two equivalent facet-inducing inequalities of the ATSP-polytope such that*

$$\beta_{uv} = \lambda\alpha_{uv} + y_u^{out} + y_v^{in} \qquad \textit{for each } uv \in E$$

and

$$\beta_0 = \lambda\alpha_0 + \sum_{v \in V}(y_v^{out} + y_v^{in}).$$

Then

$$ATSP_{LP}(\beta) = \lambda ATSP_{LP}(\alpha) + \sum_{v \in V}(y_v^{out} + y_v^{in}).$$

Proof. Let x be a feasible solution of the ATSP$_{LP}$. Since λ, y^{out}, and y^{in} are fixed and $\lambda > 0$, we see that βx is minimized with respect to x if and only if αx is minimized with respect to x. Thus x is an optimal solution of the ATSP$_{LP}$ with respect to β if and only if it is an optimal solution of the ATSP$_{LP}$ with respect to α. Furthermore, the node equalities define the equality subsystem both for the ATSP-polytope and the ASEP. Therefore,

$$\text{ATSP}_{LP}(\beta) = \lambda\text{ATSP}_{LP}(\alpha) + \sum_{v \in V}(y_v^{\text{out}} + y_v^{\text{in}}).$$

□

Now, for convenience, we define

$$\epsilon(y) = \sum_{v \in V}(y_v^{\text{out}} + y_v^{\text{in}}).$$

Then

$$\begin{aligned}\text{Gap}(\beta) &= \frac{\text{ATSP}(\beta)}{\text{ATSP}_{LP}(\beta)} \\ &= \frac{\beta_0}{\lambda\text{ATSP}_{LP}(\alpha) + \epsilon(y)} \\ &= \frac{\lambda\alpha_0 + \epsilon(y)}{\lambda\text{ATSP}_{LP}(\alpha) + \epsilon(y)}.\end{aligned}$$

We are primarily interested in the integrality gaps of metric cost functions. Hence, we would like to further investigate under what conditions β is metric.

Proposition 6.1.2. *β is metric if and only if*

$$\begin{aligned}\frac{1}{\lambda}y_u^{out} + \frac{1}{\lambda}y_v^{in} &\geq -\alpha_{uv} \qquad \textit{for each } uv \in E \\ \frac{1}{\lambda}y_w^{out} + \frac{1}{\lambda}y_w^{in} &\geq \alpha_{uv} - \alpha_{uw} - \alpha_{wv} \qquad \textit{for all distinct } u, v, w \in V.\end{aligned}$$

Proof. By definition, β is metric if and only if

$$\beta_{uv} \geq 0 \qquad \text{for each } uv \in E \tag{6.1.1}$$

and

$$\beta_{uw} + \beta_{wv} \geq \beta_{uv} \qquad \text{for all distinct } u, v, w \in V. \tag{6.1.2}$$

From equation (6.1.1) we have

$$\begin{aligned}\lambda\alpha_{uv} + y_u^{\text{out}} + y_v^{\text{in}} &\geq 0 \\ \frac{1}{\lambda}y_u^{\text{out}} + \frac{1}{\lambda}y_v^{\text{in}} &\geq -\alpha_{uv}.\end{aligned}$$

From equation (6.1.2) we have

$$\lambda\alpha_{uw} + y_u^{\text{out}} + y_w^{\text{in}} + \lambda\alpha_{wv} + y_w^{\text{out}} + y_v^{\text{in}} \geq \lambda\alpha_{uv} + y_u^{\text{out}} + y_v^{\text{in}}$$

and so

$$\begin{aligned} \lambda\alpha_{uw} + y_w^{\text{in}} + \lambda\alpha_{wv} + y_w^{\text{out}} &\geq \lambda\alpha_{uv} \\ \frac{1}{\lambda}y_w^{\text{out}} + \frac{1}{\lambda}y_w^{\text{in}} &\geq \alpha_{uv} - \alpha_{uw} - \alpha_{wv}. \end{aligned}$$

□

Lemma 6.1.3. *Let β be metric. Define*

$$\beta' = \frac{1}{\lambda}\beta,\ \beta'_0 = \frac{1}{\lambda}\beta.$$

Then $\beta' x \geq \beta'_0$ is equivalent to $\alpha x \geq \alpha_0$, β' is metric, and

$$Gap(\beta') = Gap(\beta).$$

Proof. Define $\lambda' = 1$, $(y^{\text{in}})' = \frac{1}{\lambda}y^{\text{in}}$, and $(y^{\text{out}})' = \frac{1}{\lambda}y^{\text{out}}$. Then the result follows trivially. □

Hence, in examining the integrality gaps of metric cost functions which come from facet-inducing inequalities equivalent to some $\alpha x \geq \alpha_0$, we can restrict our attention to just those which have $\lambda = 1$.

Consider the following linear program.

$$\begin{array}{llcll} (\star) & \text{minimize} & \sum_{v\in V}(y_v^{\text{out}} + y_v^{\text{in}}) & & \\ & \text{subject to:} \quad y_u^{\text{out}} + y_v^{\text{in}} & \geq & -\alpha_{uv} & \text{for all } uv \in E \\ & \quad y_w^{\text{out}} + y_w^{\text{in}} & \geq & \alpha_{uv} - \alpha_{uw} - \alpha_{wv} & \text{for all distinct } u, v, w \in V. \end{array}$$

Theorem 6.1.4. *Let $\alpha x \geq \alpha_0$ be a facet-inducing inequality of the ATSP-polytope. Let y^{in}, y^{out} be an optimal solution to the linear program to $(\star)$. Define $\beta_{uv} = \alpha_{uv} + y_u^{out} + y_v^{in}$ for each $uv \in E$ and $\beta_0 = \alpha_0 + \epsilon(y)$. Then $\beta x \geq \beta_0$ is a facet-inducing inequality of the ATSP-polytope and β achieves the maximum integrality gap among all metric cost functions which form the coefficients of a facet-inducing inequality equivalent to $\alpha x \geq \alpha_0$.*

Proof. Clearly $\beta x \geq \beta_0$ is equivalent to $\alpha x \geq \alpha_0$ by taking $\lambda = 1$. Also substituting $\lambda = 1$ into the result of Proposition 6.1.2 we see that β is metric. Furthermore, by Corollary 6.1.3 we see that in finding a cost function which maximizes the gap, we can restrict our attention to those that have $\lambda = 1$. In this case, let β' be metric where $(\beta')_{uv} = \alpha_{uv} + (y^{\text{out}})'_u + (y^{\text{in}})'_v$ for each $uv \in E$. Then

$$\begin{aligned} \text{Gap}(\beta') &= \frac{\alpha_0 + \epsilon(y')}{\text{ATSP}_{LP}(\alpha) + \epsilon(y')} \\ &= 1 + \frac{\alpha_0 - \text{ATSP}_{LP}(\alpha)}{\text{ATSP}_{LP}(\alpha) + \epsilon(y')}. \end{aligned}$$

However, $\text{ATSP}_{LP}(\beta') = \text{ATSP}_{LP}(\alpha) + \epsilon(y')$ and we know $\beta' \geq 0$ since β' is metric. Furthermore, $\alpha_0 \geq \text{ATSP}_{LP}(\alpha)$ since $\text{ATSP}(\alpha) \geq \text{ATSP}_{LP}(\alpha)$. Thus $\text{Gap}(\beta')$ is maximized when

$$\frac{\alpha_0 - \text{ATSP}_{LP}(\alpha)}{\text{ATSP}_{LP}(\alpha) + \epsilon(y')}$$

is maximized or equivalently when $\epsilon(y')$ is minimized. The result follows. □

Thus we can turn our attention to the linear program ($\star$) to find the largest possible integrality gap obtained from some facet-inducing inequality. Our goal is to find an optimal solution to this linear program.

To this end, we define

$$\alpha_{ww} = \max_{\substack{u,v \in V\setminus\{w\} \\ u \neq v}} \{\alpha_{uv} - \alpha_{uw} - \alpha_{wv}\}$$

and

$$\phi_w = \sum_{h \in V\setminus\{w\}} \alpha_{hw} - \sum_{h \in V\setminus\{w\}} \alpha_{wh} \qquad \text{for each } w \in V.$$

Let

$$y_v^{\text{out}} = \frac{1}{2}\alpha_{vv} + \frac{1}{2n}\phi_v, \tag{6.1.3}$$

$$y_v^{\text{in}} = \frac{1}{2}\alpha_{vv} - \frac{1}{2n}\phi_v. \tag{6.1.4}$$

We now show that this is an optimal solution to the linear program.

Lemma 6.1.5. *The values y^{out} and y^{in} as given in (6.1.3) and (6.1.4) are a feasible solution to the linear program* ($\star$).

Proof. Let $w \in V$.

$$\begin{aligned}
y_w^{\text{out}} + y_w^{\text{in}} &= \frac{1}{2}\alpha_{ww} + \frac{1}{2n}\phi_w + \frac{1}{2}\alpha_{ww} - \frac{1}{2n}\phi_w \\
&= \alpha_{ww} \\
&= \max_{\substack{u, v \in V\setminus\{w\} \\ u \neq v}} \{\alpha_{uv} - \alpha_{uw} - \alpha_{wv}\} \\
&\geq \alpha_{uv} - \alpha_{uw} - \alpha_{wv} \qquad \text{for any distinct } u, v \in V\setminus\{w\}.
\end{aligned}$$

Now, let $uv \in E$.

$$\begin{aligned}
y_u^{\text{out}} + y_v^{\text{in}} &= \frac{1}{2}\alpha_{uu} + \frac{1}{2}\alpha_{vv} + \frac{1}{2n}\phi_u - \frac{1}{2n}\phi_v \\
&= \frac{1}{2n}\left(n\alpha_{uu} + n\alpha_{vv} + \sum_{h\in V\setminus\{u\}} \alpha_{hu} - \sum_{h\in V\setminus\{u\}} \alpha_{uh} - \sum_{h\in V\setminus\{v\}} \alpha_{hv} + \sum_{h\in V\setminus\{v\}} \alpha_{vh}\right) \\
&= \frac{1}{2n}\left(n\alpha_{uu} + n\alpha_{vv} + 2\alpha_{vu} - 2\alpha_{uv} + \sum_{h\in V\setminus\{u,v\}} (\alpha_{hu} - \alpha_{uh} - \alpha_{hv} + \alpha_{vh})\right) \\
&= \frac{1}{2n}(2\alpha_{uu} + 2\alpha_{vv} + 2\alpha_{vu} + 2\alpha_{uv} - 2n\alpha_{uv}) \\
&\quad + \frac{1}{2n}\sum_{h\in V\setminus\{u,v\}} (\alpha_{uu} - (\alpha_{hv} - \alpha_{hu} - \alpha_{uv})) \\
&\quad + \frac{1}{2n}\sum_{h\in V\setminus\{u,v\}} (\alpha_{vv} - (\alpha_{uh} - \alpha_{uv} - \alpha_{vh})) \\
&\geq \frac{1}{2n}(2\alpha_{uu} + 2\alpha_{vv} + 2\alpha_{vu} + 2\alpha_{uv} - 2n\alpha_{uv}) \\
&= -\alpha_{uv} + \frac{1}{2n}(2\alpha_{uu} + 2\alpha_{vv} + 2\alpha_{vu} + 2\alpha_{uv}) \\
&= -\alpha_{uv} + \frac{1}{2n}(\alpha_{uu} - (\alpha_{hv} - \alpha_{hu} - \alpha_{uv})) + \frac{1}{2n}(\alpha_{uu} - (\alpha_{vh} - \alpha_{vu} - \alpha_{uh})) \\
&\quad + \frac{1}{2n}(\alpha_{vv} - (\alpha_{uh} - \alpha_{uv} - \alpha_{vh})) + \frac{1}{2n}(\alpha_{vv} - (\alpha_{hu} - \alpha_{hv} - \alpha_{vu})) \\
&\quad (\text{ for any } h \in V\setminus\{u, v\}) \\
&\geq -\alpha_{uv}.
\end{aligned}$$

□

Theorem 6.1.6. *The values y^{out} and y^{in} as given in (6.1.3) and (6.1.4) are an optimal solution to the linear program ($\star$) and the optimal value is*

$$\sum_{v \in V} \alpha_{ww}.$$

Proof. From the constraints of our linear program, we see that for a given $w \in V$ we must satisfy

$$y_w^{\text{out}} + y_w^{\text{in}} \geq \alpha_{uv} - \alpha_{uw} - \alpha_{wv},$$

for each $u, v \in V \backslash \{w\}$ such that $v \neq u$. Hence

$$y_w^{\text{out}} + y_w^{\text{in}} \geq \alpha_{ww}$$

is a valid inequality for the linear program for each $w \in V$. Thus

$$\sum_{w \in V} (y_w^{\text{out}} + y_w^{\text{in}}) \geq \sum_{v \in V} \alpha_{ww},$$

and so we have a lower bound on the optimal value of the linear program.

For our choice of y^{out} and y^{in} as given in (6.1.3) and (6.1.4) we see that

$$y_w^{\text{out}} + y_w^{\text{in}} = \alpha_{ww}$$

for each $w \in V$. Thus, our choice y^{out} and y^{in} attains the above lower bound on the optimal value and hence is an optimal solution. □

Now that we have an optimal solution to the linear program, we can define β and β_0. Specifically,

$$\beta_{uv} = \alpha_{uv} + \frac{1}{2}(\alpha_{uu} + \alpha_{vv}) + \frac{1}{2n}(\phi_u - \phi_v)$$

and

$$\beta_0 = \alpha_0 + \sum_{w \in V} \alpha_{ww}.$$

Our construction of β and β_0 is similar to the so called *canonical form* of a facet-inducing inequality as introduced by Balas and Fischetti in [4]. Our construction is slightly simpler and is specially tailored to easily compute the resulting integrality gap.

We can now compute Gap(β) as a function of α and α_0 only, namely,

$$\text{Gap}(\beta) = \frac{\alpha_0 + \sum_{v \in V} \alpha_{vv}}{\text{ATSP}_{LP}(\alpha) + \sum_{v \in V} \alpha_{vv}}.$$

Recall, however, the importance of β. It is a metric cost function which forms the coefficients of a facet-inducing inequality equivalent to $\alpha x \geq \alpha_0$ which has a maximum integrality gap. However, this maximum value is unique (even if different metric cost functions forming facet-inducing inequalities equivalent to $\alpha x \geq \alpha_0$ might attain the maximum). Hence, this value of the maximum integrality gap is a unique measure for the facet. Therefore, let us define the *strength* of a facet F of the ATSP-polytope to be

$$\text{Str}(F) = \frac{\alpha_0 + \sum_{v \in V} \alpha_{vv}}{\text{ATSP}_{LP}(\alpha) + \sum_{v \in V} \alpha_{vv}}$$

where $\alpha x \geq \alpha_0$ is any inequality which induces the facet F. Hence, given any facet inducing inequality of a given facet F, we can compute Str(F). A similar approach is used by Goemans in [39] to compute the integrality gap of cost functions constructed from various facet-inducing inequalities of the Graphical Travelling Salesman Polyhedron to the (Symmetric) Subtour Elimination Polytope with the vertex equalities relaxed to inequality. However, both polyhedra are full-dimensional (and hence had no equality set) so it does not make sense for Goemans to be concerned with equivalent forms of the inequalities. Interestingly, their facets are always in tight triangular form ($\alpha_{vv} = 0$ for all $v \in V$) since Naddef and Rinaldi [70] proves that facets of the STSP-polytope are facets of the Graphical Travelling Salesman Polyhedron only if they are in tight triangular form.

In the remaining sections of this chapter, we examine several well-known facets of the ATSP-polytope and use the theory from this section to compute the strength of each.

6.2 Facets of the ATSP-polytope

In 1989, Bartels and Bartels [6] gave a complete description of all the facets of the ATSP-polytope on 5 nodes. In total, there are 390 which they listed in six different classes. In 1995, Euler and Le Verge [30] gave a complete description of the facets of the ATSP-polytope on 6 nodes. Astoundingly, there are 319,015 different facets.

Among the most well-known facet-inducing inequalities of the ATSP-polytope we have the nonnegativity constraints

$$x_{uv} \geq 0 \qquad \text{for all } uv \in E$$

and the cut constraints

$$x(\delta(S)) \geq 1 \qquad \text{for all } \emptyset \subset S \subset V.$$

Notice, however, that these are also facet-inducing inequalities of the ASEP. Thus, for either of these classes of facet-inducing inequalities in the form $\alpha x \geq \alpha_0$ we have that $\text{ATSP}_{LP}(\alpha) = \alpha_0$. Therefore

$$\text{Str}(F) = 1$$

for any facet F induced by an inequality in either of these classes. We now continue our investigation into more complicated facets of the ATSP-polytope.

6.3 Clique Tree Inequalities

Clique Tree Inequalities are facet-inducing inequalities of the ATSP-polytope which were first introduced by Grötschel and Pulleyblank [47] as facets of the STSP-polytope. Fischetti [32] later proved that the asymmetric version of these inequalities produces facets of the ATSP-polytope for all $n \geq 7$. In order define this important class of facet-inducing inequalities, we first define a *intersection graph*. Given a collection of subsets of a set V, we create a vertex in the intersection graph for each subset and two vertices in the intersection graph are adjacent if and only if the corresponding subsets have a non-empty intersection.

Let $H_1, \ldots, H_r$ be disjoint subsets of V which we call *handles* and let $T_1, \ldots, T_s$ be disjoint subsets of V which we call *teeth*. Let h_i be the number of teeth intersecting the handle H_i and let t_j be the number of handles intersecting the tooth T_j. We say that this collection of subsets is a *clique tree* if

1. $r \geq 1$,
2. $s \geq 3$ is odd,
3. every tooth has a node not in any handle,
4. $h_i \geq 3$ is odd for each $1 \leq i \leq r$, and
5. the intersection graph of all these subsets is a tree.

The associated *clique tree inequality* is

$$-\sum_{i=1}^{r} x(\gamma(H_i)) - \sum_{j=1}^{s} x(\gamma(T_j)) \geq -\sum_{i=1}^{r} |H_i| - \sum_{j=1}^{s}(|T_j| - t_j) + \frac{s+1}{2}$$

(notice that we have just multiplied the most popular version of this inequality by -1 so that the inequality sign points in the same direction as it does in the theory at the beginning of this chapter). Figure 6.1 shows a picture of a clique tree inequality corresponding to a clique tree with one handle and three teeth. The circle represents the handle and the rectangles represent the teeth. Thin arcs have coefficients of -1 in the inequality while thick arcs have coefficients of -2. The right hand side of the inequality is -6.

For convenience, let $\alpha x \geq \alpha_0$ denote the clique tree inequality. In order to calculate the strength of the corresponding facet, we need to calculate $\text{ATSP}_{LP}(\alpha)$. We do this by using the results from the symmetric case as presented in Boyd and Pulleyblank [15]. We provide the necessary details to prove these results for the asymmetric case.

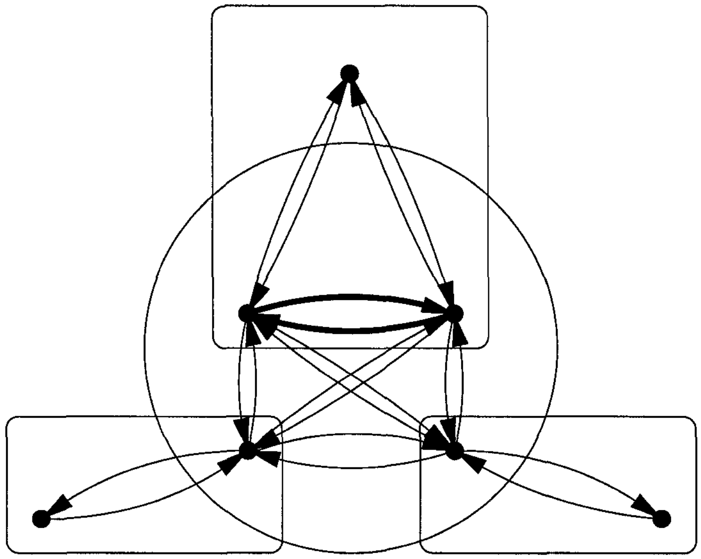

Figure 6.1: A clique tree inequality

For a given clique tree we say that a given tooth T_j is a *pendent tooth* if $t_j = 1$. Otherwise, we say that T_j is a *nonpendent tooth.* Let q denote the number of nonpendent teeth. For any tooth T_j, let $\tilde{T}_j$ denote the set of nodes of T_j which are in no handle. For any handle H_i let $\tilde{H}_i$ be the set of nodes of H_i which are in no tooth (note that $\tilde{H}_i$ may be empty). Finally, let $\tilde{V}$ denote the set of nodes of V which are in no tooth nor in any handle.

Theorem 6.3.1.

$$ATSP_{LP}(\alpha) = -\sum_{i=1}^{r}|H_i| - \sum_{j=1}^{s}(|T_j| - t_j) + \frac{s+1}{2} - \frac{q+1}{2}.$$

Proof. Boyd and Pulleyblank [15] show that in the symmetric case,

$$-\sum_{i=1}^{r} x(\gamma(H_i)) - \sum_{j=1}^{s} x(\gamma(T_j)) \geq -\sum_{i=1}^{r}|H_i| - \sum_{j=1}^{s}(|T_j| - t_j) + \frac{s+1}{2} - \frac{q+1}{2}$$

is a valid equality for the Subtour Elimination Polytope. Furthermore, they provide a specific point of the SEP to show that this inequality is tight.

Now, by a comment of Grötschel and Padberg [46], any valid equality $\beta' x' \geq \beta'_0$ for the SEP can induce a valid inequality $\beta x \geq \beta_0$ for the ASEP by the transformation

$\beta_{uv} = \beta_{vu} = \beta'_{uv}$ for each unordered pair of distinct $u, v \in V$ and $\beta_0 = \beta'_0$. Better yet, if we have a point x' of the SEP such that $\beta' x' = \beta'_0$ then we can set $x_{uv} = x_{vu} = \frac{1}{2}x'_{uv}$ for each unordered pair of distinct $u, v \in V$ and we get that x is a feasible point of the ASEP and $\beta x = \beta_0$.

Therefore,

$$-\sum_{i=1}^{r} x(\gamma(H_i)) - \sum_{j=1}^{s} x(\gamma(T_j)) \geq -\sum_{i=1}^{r} |H_i| - \sum_{j=1}^{s}(|T_j| - t_j) + \frac{s+1}{2} - \frac{q+1}{2}$$

is also a valid inequality for the ASEP which is tight for some x in the ASEP. □

Next, we must compute $\sum_{w \in V} \alpha_{ww}$ for the clique tree inequality.

Proposition 6.3.2.

$$\alpha_{ww} = \begin{cases} 1 & \textit{if } w \in \tilde{H}_i \textit{ for some } 1 \leq i \leq r \\ 1 & \textit{if } w \in \tilde{T}_j \textit{ for some } 1 \leq j \leq s \\ 2 & \textit{if } w \in H_i \cap T_j \textit{ for some } 1 \leq i \leq r \textit{ and } 1 \leq j \leq s \\ 0 & \textit{otherwise} \end{cases}$$

and so

$$\sum_{w \in V} \alpha_{ww} = \sum_{i=1}^{r} |H_i| + \sum_{j=1}^{s} |T_j|.$$

Proof. For this proof, we consider four separate cases.
Case 1: $w \in \tilde{H}_i$ for some $1 \leq i \leq r$.

Then the only arcs with non-zero α-values incident to w have have their other endpoint in H_i and their α-value is -1. Hence, if we choose $u, v \in H_i$ then we get

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 1.$$

If one of u and v are not in H_i then clearly

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} \leq 1$$

and the result follows.

Case 2: $w \in \tilde{T}_j$ for some $1 \leq j \leq s$.

This case is identical to Case 1 (replacing H_i with T_j) except that we need to deal with the possibility that $|T_j| = 2$ and so it can't be that $u, v \in T_j$. In this case, we chose $u \in T_j$ and $v \notin T_j$ and we get that

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 1$$

so the result follows.

Case 3: $w \in H_i \cap T_j$ for some $1 \leq i \leq r$ and $1 \leq j \leq s$.

If we choose $u \in T_j - H_i$ and $v \in H_i - T_j$ then we get that

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 2.$$

To get a larger sum, we would need to choose either u or v to be in $H_i \cap T_j$. Suppose, without loss of generality, we have $u \in H_i \cap T_j$. Then $\alpha_{uv} = \alpha_{wv}$ and so

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = -\alpha_{uw} = 2$$

which is not a larger sum.

Case 4: $w \in \tilde{V}$.

Then every arc incident to w has an α-value of 0. By choosing some $1 \leq i \leq r$ and $1 \leq j \leq s$ and choosing any $u \in H_i - T_j$ and any $v \in T_j - H_i$ we get that

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 0$$

and the result follows.

Therefore, by adding up all the α_{ww} values, we get

$$\sum_{w \in V} \alpha_{ww} = \sum_{i=1}^{r} |H_i| + \sum_{j=1}^{s} |T_j|.$$

□

Now we are ready to compute the strength of the facet F induced by the clique tree inequality.

Theorem 6.3.3.

$$Str(F) = \frac{2r + 3s - 1}{2r + 3s - q - 2}.$$

Proof. From Theorem 6.3.1 we have that

$$\text{ATSP}_{LP}(\alpha) = -\sum_{i=1}^{r} |H_i| - \sum_{j=1}^{s}(|T_j| - t_j) + \frac{s+1}{2} - \frac{q+1}{2}$$

and from Proposition 6.3.2 we have that

$$\sum_{w \in V} \alpha_{ww} = \sum_{i=1}^{r} |H_i| + \sum_{j=1}^{s} |T_j|.$$

Thus,

$$\begin{aligned} \text{Str}(F) &= \frac{(-\sum_{i=1}^{r} |H_i| - \sum_{j=1}^{s}(|T_j| - t_j) + \frac{s+1}{2}) + (\sum_{i=1}^{r} |H_i| + \sum_{j=1}^{s} |T_j|)}{(-\sum_{i=1}^{r} |H_i| - \sum_{j=1}^{s}(|T_j| - t_j) + \frac{s+1}{2} - \frac{q+1}{2}) + (\sum_{i=1}^{r} |H_i| + \sum_{j=1}^{s} |T_j|)} \\ &= \frac{\sum_{j=1}^{s} t_j + \frac{s+1}{2}}{\sum_{j=1}^{s} t_j + \frac{s+1}{2} - \frac{q+1}{2}} \\ &= \frac{2\sum_{j=1}^{s} t_j + s + 1}{2\sum_{j=1}^{s} t_j + s - q}. \end{aligned}$$

However, t_j is exactly the degree of the vertex in the intersection graph corresponding to the tooth T_j. Furthermore, this intersection graph is bipartite with the vertices corresponding to the teeth forming one part of the bipartition. Hence $\sum_{j=1}^{s} t_j$ is the number of edges in the intersection graph. Now, the intersection graph has $r+s$ vertices and is a tree so it has $r + s - 1$ edges. Thus, $\sum_{j=1}^{s} t_j = r + s - 1$. Therefore,

$$\begin{aligned} \text{Str}(F) &= \frac{2(r + s - 1) + s + 1}{2(r + s - 1) + s - q} \\ &= \frac{2r + 3s - 1}{2r + 3s - q - 2}. \end{aligned}$$

□

Before we proceed to a simpler upper bound on the strength of the facet induced by a clique tree inequality, we prove a necessary result about clique trees which also appears in Goemans [39].

Lemma 6.3.4. *For any clique tree, $s \geq 2r+1$ and $q \leq r-1$.*

Proof. To prove the results we proceed by induction on r. If $r = 1$ then $q = 0$ and, by the definition of a clique tree, $s \geq 3$ and the results follow. Now suppose that the results hold for any clique tree with fewer than r handles where $r \geq 2$.

Consider a clique tree with exactly r handles, s teeth, and q nonpendent teeth. Since $r \geq 2$, we know that, since the intersection graph of the subsets that make up the clique tree form a tree and all the vertices of the tree of degree 1 correspond to teeth, there must be a handle, H, which intersects exactly one nonpendent tooth, T. Suppose that H intersects exactly t teeth (so $t \geq 3$). Then, by removing H and all $t-1$ pendent teeth which intersect H in the clique tree, we get a new clique tree with precisely $r-1$ handles, $s-(t-1)$ teeth, and $q-1$ nonpendent teeth (since T is a pendent tooth of the new clique tree but all other nonpendent teeth of the old clique tree remain nonpendent teeth of the new clique tree).

Thus, by our first inductive hypothesis,

$$s-(t-1) \geq 2(r-1)+1.$$

Hence,

$$\begin{aligned} s-t+1 &\geq 2r-2+1 \\ s &\geq 2r+t-2. \end{aligned}$$

But, $t \geq 3$, so we have $s \geq 2r+1$ and the first result follows by induction.

By our second inductive hypothesis, we have

$$q-1 \leq (r-1)-1$$

and so $q \leq r-1$ and the second result follows by induction. □

Corollary 6.3.5. *For any facet F induced by a clique tree inequality,*

$$Str(F) \leq \frac{8}{7}.$$

Proof. From Theorem 6.3.3,

$$\begin{aligned} \text{Str}(F) &= \frac{2r+3s-1}{2r+3s-q-2} \\ &= 1+\frac{q+1}{2r+3s-q-2} \end{aligned}$$

Now, by Lemma 6.3.4, $s \geq 2r+1$ and $q \leq r-1$. Thus,

$$\begin{aligned} \text{Str}(F) &\leq 1+\frac{q+1}{2r+3(2r+1)-q-2} \\ &= 1+\frac{q+1}{8r-q+1} \\ &\leq 1+\frac{(r-1)+1}{8r-(r-1)+1} \\ &= 1+\frac{r}{7r+2} \\ &= \frac{8r+2}{7r+2} \\ &= \frac{8}{7}-\frac{2}{7(7r+2)}. \end{aligned}$$

□

We now turn our attention to some facets of the ATSP-polytope which are not obtained from facets of the STSP-polytope. Grötschel [43] along with Padberg [45] introduce several new classes of valid inequalities for the ATSP. Specifically, the D_k^+, D_k^-, T_k, $C2$, and $C3$ inequalities. In their papers, they prove that the T_k inequalities are facet-inducing. Later, Fischetti [31] proves that the D_k^+, D_k^-, $C2$, and $C3$ inequalities in each of these families are facet-inducing. We now examine each class of facet-inducing inequalities in turn and compute their strength.

6.4 D_k^+ and D_k^- Inequalities

The D_k^- inequality is defined as follows. Let $v_1, v_2, \ldots, v_k$ be distinct nodes of V such that $3 \leq k \leq n-1$. Then the D_k^- inequality is

$$-\sum_{i=1}^{k-1} x_{v_i v_{i+1}} - x_{v_k v_1} - 2\sum_{i=2}^{k-1} x_{v_i v_1} - \sum_{i=3}^{k-1}\sum_{j=2}^{i-1} x_{v_i v_j} \geq -k+1.$$

Note, that our version of this inequality is obtained simply by multiplying the typical form of this inequality by -1 so that our inequality sign points in the right direction. A picture of this inequality is given in Figure 6.2 where the node v_i is simply labelled with i. The dotted arcs and solid arcs have coefficients of -1 and -2 respectively in the D_k^- inequality. Essentially, the α-values of the arcs of the dicycle $(v_1, v_2, \ldots, v_k)$ are all -1. Also, for any arc v_iv_j such that $2 \leq j < i \leq k-1$ we have $\alpha_{v_iv_j} = -1$. Lastly, we have $\alpha_{v_iv_1} = -2$ for each $2 \leq i \leq k-1$. For simplicity, we use $\alpha x \geq -k+1$ to denote this D_k^- inequality.

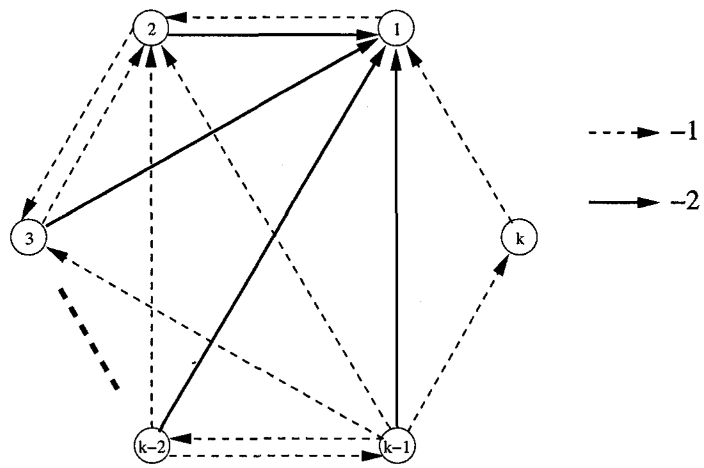

Figure 6.2: D_k^-

We can compute in a straightforward manner that

$$\alpha_{vv} = \begin{cases} 2 & \text{if } v \in \{v_1, \ldots, v_{k-1}\} \\ 1 & \text{if } v = v_k \\ 0 & \text{otherwise} \end{cases}$$

and so

$$\sum_{v \in V} \alpha_{vv} = 2k - 1.$$

Next, we compute $\text{ATSP}_{LP}(\alpha)$ by producing a feasible ASEP point and proving its optimality by using the dual of this linear program. Recall that a dual solution has a value d_S for each $\emptyset \subset S \subset V$ and an ordered pair $(y_v^{\text{in}}, y_v^{\text{out}})$ for each $v \in V$ which obey the constraints

$$\begin{aligned} y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid uv \in \delta(S)) &\leq \alpha_{uv} \text{ for each } uv \in E \\ d_S &\geq 0 \text{ for each } \emptyset \subset S \subset V. \end{aligned}$$

Theorem 6.4.1.

$$ATSP_{LP}(\alpha) = -\frac{k^2 - k - 1}{k - 1}.$$

Proof. Let $u_1, \ldots, u_r$ be the nodes of $V \backslash \{v_1, \ldots, v_k\}$. Define an ASEP point x as follows.

$$x_{uv} = \begin{cases} \frac{k-2}{k-1} & \text{if } uv = v_i v_{i+1} \text{ for some } 1 \leq i \leq k-1 \\ & \text{or } uv = v_k u_1 \\ & \text{or } uv = u_i u_{i+1} \text{ for some } 1 \leq i \leq r-1 \\ \frac{1}{k-1} & \text{if } uv = u_r v_i \text{ for some } 2 \leq i \leq k-1 \\ & \text{or } uv = v_i v_1 \text{ for some } 2 \leq i \leq k \\ & \text{or } uv = u_i u_{i-1} \text{ for some } 2 \leq i \leq r \\ & \text{or } uv = u_1 v_k \\ & \text{or } uv = v_1 u_r \\ 0 & \text{otherwise.} \end{cases}$$

A picture of x when $k = 5$ and $r = 2$ is given in Figure 6.3. The dotted arcs have an x-value of $\frac{1}{4}$ and the solid arcs have an x-value of $\frac{3}{4}$.

It is not difficult to see that x is in fact a point of the ASEP due to the fact that

- the arcs of value $\frac{k-2}{k-1}$ induce a directed Hamiltonian (v_1, u_r)-dipath,
- there are $k - 1$ internally node-disjoint (u_r, v_1)-dipaths each consisting only of arcs of value $\frac{1}{k-1}$, and
- each node is in some dicycle containing v_1 and u_r consisting entirely of arcs of value $\frac{1}{k-1}$.

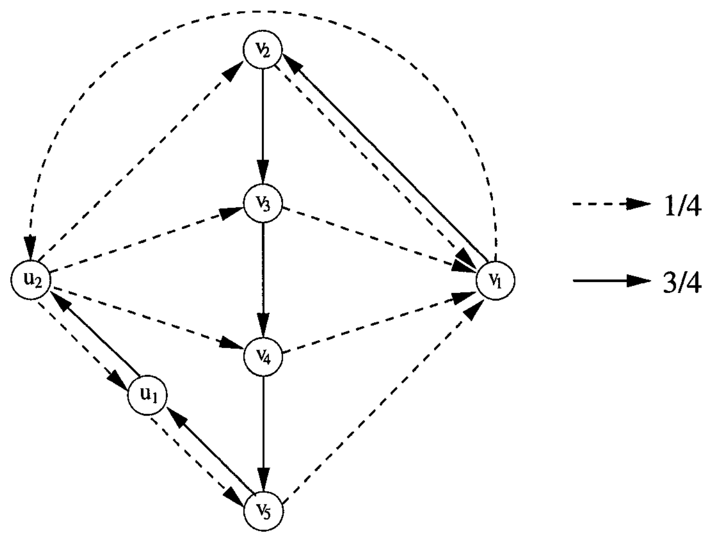

Figure 6.3: x when $k = 5$ and $r = 2$

The total cost of x is

$$\begin{aligned}
\alpha x &= \sum_{uv \in E} \alpha_{uv} x_{uv} \\
&= \alpha_{v_k v_i} x_{v_i v_{i+1}} + \sum_{i=1}^{k-1} \alpha_{v_i v_{i+1}} x_{v_i v_{i+1}} + \sum_{i=2}^{k-1} \alpha_{v_i v_1} x_{v_i v_1} \\
&= (-1)\frac{1}{k-1} + \sum_{i=1}^{k-1}(-1)\frac{k-2}{k-1} + \sum_{i=2}^{k-1}(-2)\frac{1}{k-1} \\
&= -\frac{1}{k-1} - \frac{(k-1)(k-2)}{k-1} - \frac{2(k-2)}{k-1} \\
&= -\frac{k^2-k-1}{k-1}.
\end{aligned}$$

To prove that x is in fact optimal with respect to α, we use the dual of ATSP_{LP} and present a feasible dual solution whose objective value is exactly αx.

First we define $V_i = \{v_1, v_2, \ldots, v_i\}$ for each $2 \leq i \leq k$. Then set

$$d_S = \begin{cases} \frac{1}{k-1} & \text{if } S = V_i \text{ for some } 2 \leq i \leq k \\ 0 & \text{otherwise} \end{cases}$$

and

$$(y_v^{\text{in}}, y_v^{\text{out}}) = \begin{cases} (-\frac{k-2}{k-1}, -1) & \text{if } v = v_1 \\ (0, -\frac{k}{k-1}) & \text{if } v = v_i \text{ for some } 2 \le i \le k-1 \\ (0, -\frac{1}{k-1}) & \text{if } v = v_k \\ (0,0) & \text{otherwise.} \end{cases}$$

To prove that we have a feasible dual solution, first note that $y_v^{\text{in}} \le 0$ for each $v \in V$, $y_v^{\text{out}} \le 0$ for each $v \in V$, and $\sum_{\emptyset \subset S \subset V} d_S = 1$. Hence, if $uv \in E$ such that $uv \notin \delta(V_i)$ for any $2 \le i \le k$ then

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid uv \in \delta(S)) \le 0.$$

If $u = v_i$ for some $1 \le i \le k-1$ then $y_u^{\text{out}} \le -1$ and $\sum_{\emptyset \subset S \subset V}(d_S \mid uv \in \delta(S)) \le 1$ so

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid uv \in \delta(S)) \le 0.$$

Lastly, if $u = v_k$ then $y_u^{\text{out}} = -\frac{1}{k-1}$ and $u \notin V_{k-1}$ so $\sum_{\emptyset \subset S \subset V}(d_S \mid uv \in \delta(S)) \le \frac{1}{k-1}$. Hence,

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid uv \in \delta(S)) \le 0$$

and so for all $uv \in E$ we have

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid uv \in \delta(S)) \le 0.$$

Now, for $2 \le i \le k-1$ we have $y_{v_i}^{\text{out}} = -\frac{k}{k-1}$, $y_{v_{i+1}}^{\text{in}} = 0$, and $\sum_{\emptyset \subset S \subset V}(d_S \mid v_i v_{i+1} \in \delta(S)) = d_{V_i} = \frac{1}{k-1}$. Thus,

$$y_{v_i}^{\text{out}} + y_{v_{i+1}}^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid v_i v_{i+1} \in \delta(S)) = -1 = \alpha_{v_i v_{i+1}}.$$

Also $y_{v_1}^{\text{out}} = -1$, $y_{v_2}^{\text{in}} = 0$, and $\sum_{\emptyset \subset S \subset V}(d_S \mid v_1 v_2 \in \delta(S)) = 0$. Hence,

$$y_{v_1}^{\text{out}} + y_{v_2}^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid v_1 v_2 \in \delta(S)) = -1 = \alpha_{v_1 v_2}.$$

Additionally, $y_{v_k}^{\text{out}} = -\frac{1}{k-1}$, $y_{v_1}^{\text{in}} = -\frac{k-2}{k-1}$, and $\sum_{\emptyset\subset S\subset V}(d_S \mid v_k v_1 \in \delta(S)) = 0$. Hence,

$$y_{v_k}^{\text{out}} + y_{v_1}^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid v_k v_1 \in \delta(S)) = -1 = \alpha_{v_k v_1}.$$

Next, for each $3 \le i \le k-1$ and $2 \le j \le i-1$ we have $y_{v_i}^{\text{out}} = -\frac{k}{k-1}$, $y_{v_j}^{\text{in}} = 0$, and $\sum_{\emptyset\subset S\subset V}(d_S \mid v_i v_j \in \delta(S)) = 0$. Hence,

$$y_{v_i}^{\text{out}} + y_{v_j}^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid v_i v_j \in \delta(S)) = -\frac{k}{k-1} \le -1 = \alpha_{v_i v_j}.$$

Lastly, for each $2 \le i \le k-1$ we have $y_{v_i}^{\text{out}} = -\frac{k}{k-1}$, $y_{v_1}^{\text{in}} = -\frac{k-2}{k-1}$, and $\sum_{\emptyset\subset S\subset V}(d_S \mid v_i v_1 \in \delta(S)) = 0$. Hence,

$$y_{v_i}^{\text{out}} + y_{v_j}^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid v_i v_j \in \delta(S)) = -2 = \alpha_{v_i v_1}.$$

Therefore, we do in fact have a feasible dual solution. Furthermore, the objective value of this dual solution is

$$\begin{aligned}\sum_{v\in V} y_v^{\text{out}} + \sum_{v\in V} y_v^{\text{in}} + \sum_{\emptyset\subset S\subset V} d_S &= -\frac{(k-1)k}{k-1} - \frac{k-2}{k-1} + 1 \\ &= -\frac{k^2-k-1}{k-1}.\end{aligned}$$

This is the exact same as the objective value of our primal solution so we know that x is optimal with respect to α. Hence,

$$\text{ATSP}_{LP}(\alpha) = -\frac{k^2-k-1}{k-1}.$$

□

Let F denote the facet induced by the D_k^- inequality. Now that we know both $\sum_{v\in V} \alpha_{vv}$ and $\text{ATSP}_{LP}(\alpha)$ we are ready to compute the strength of F.

Corollary 6.4.2. *For each $k \ge 3$ we have*

$$Str(F) = \frac{k^2-k}{k^2-2k+2}$$

and so for all $k \ge 3$

$$Str(F) \le \frac{6}{5}$$

Proof. As noted at the beginning of this section, we see that $\sum_{v \in V} \alpha_{vv} = 2k - 1$ and from Theorem 6.4.1 we have that $\text{ATSP}_{LP}(\alpha) = -\frac{k^2-k-1}{k-1}$. Thus

$$\begin{aligned}
\text{Str}(F) &= \frac{-k+1+(2k-1)}{-\frac{k^2-k-1}{k-1}+(2k-1)} \\
&= \frac{k}{\frac{k^2-2k+2}{k-1}} \\
&= \frac{k(k-1)}{k^2-2k+2} \\
&= \frac{k^2-k}{k^2-2k+2}.
\end{aligned}$$

By analyzing the derivative of the above ratio, we see that $\frac{k^2-k}{k^2-2k+2}$ is a strictly decreasing function for all $k \geq 4$. By calculating $\text{Str}(F)$ in the special cases that $k = 3$ or $k = 4$ we get that $\text{Str}(F) = \frac{6}{5}$ in either case and the result follows. □

As for the D_k^+ inequality, it is quite similar to the D_k^- inequality. Specifically, the D_k^+ inequality is

$$-\sum_{i=1}^{k-1} x_{v_i v_{i+1}} - x_{v_k v_1} - 2\sum_{i=3}^{k} x_{v_1 v_i} - \sum_{i=4}^{k}\sum_{j=3}^{i-1} x_{v_i v_j} \geq -k+1.$$

To obtain the D_k^+ inequality $\beta x \geq -k+1$ from the D_k^- inequality $\alpha x \geq -k+1$ we simply use the transformation

$$\beta_{v_i v_j} = \alpha_{v_{k-j+2} v_i}$$

where a subscript of "$k+1$" should be read as a subscript of "1". In this way, a nearly identical argument to that for the D_k^- inequality can prove that for the facet F induced by the D_k^+ inequality, we also get the result in Corollary 6.4.2.

In the next section we explore another famous facet-inducing of inequality of the ATSP obtained by adding a little extra to a cut constraint.

6.5 T_k Inequalities

The T_k inequalities are a modification of the cut constraints. Let $\emptyset \subset S \subset V$ such that $2 \leq |S| \leq n-2$, let $w \in S$, and let $p, q \in V \backslash S$. Let $k = |S|$ then the T_k

inequality is defined as

$$x(\delta(S)) - x_{pw} - x_{wq} - x_{pq} \geq 0.$$

We use this version of the inequality since it makes our calculations a bit simpler. Many authors use the form

$$x(\gamma(S)) + x_{pw} + x_{wq} + x_{pq} \leq |S|.$$

Our equivalent inequality can be obtained by subtracting certain node equalities, namely,

$$x(\gamma(S)) + x_{pw} + x_{wq} + x_{pq} - \sum_{v \in S} x(\delta(v)) \leq |S| - |S|.$$

But, $x(\gamma(S)) = \sum_{v \in S} x(\delta(v)) - x(\delta(S))$, and so

$$-x(\delta(S)) + x_{pw} + x_{wq} + x_{pq} \leq 0.$$

By multiplying both sides of this inequality by -1 we get

$$x(\delta(S)) - x_{pw} - x_{wq} - x_{pq} \geq 0.$$

If we represent this inequality as $\alpha x \geq 0$ then it is straightforward to compute

$$\alpha_{vv} = \begin{cases} 1 & \text{if } v \in \{p, q, w\} \\ 0 & \text{otherwise} \end{cases}$$

and so

$$\sum_{v \in V} \alpha_{vv} = 3.$$

Now, let F be the facet induced by the T_k inequality. In order to compute the strength of this facet, we need to compute $\mathrm{ATSP}_{LP}(\alpha)$. Again, we accomplish this by presenting a feasible point of the ASEP and proving that it is an optimal solution with respect to α by giving a feasible dual solution whose optimal value is the same.

Theorem 6.5.1. *For any k*

$$ATSP_{LP}(\alpha) = -\frac{1}{2}.$$

Proof. Let $v_1, \ldots, v_{n-2}$ be the nodes of $V \backslash \{p, q\}$ such that $w = v_1$ and $v_1, \ldots, v_k$ are the nodes of S. Define a point x such that

$$x_{uv} = \begin{cases} \frac{1}{2} & \text{if } uv = v_i v_{i+1} \text{ or } uv = v_{i+1} v_i \text{ for some } 1 \leq i \leq n-3 \\ & \text{or } uv \in \{wq, pw, pq, qp, v_{n-2}p, qv_{n-2}\} \\ 0 & \text{otherwise.} \end{cases}$$

Figure 6.4 depicts x when $k = 3$ and $n = 8$. Every arc in the diagram has an x-value of $\frac{1}{2}$ and all other arcs have an x-value of 0. The subset S is also shown.

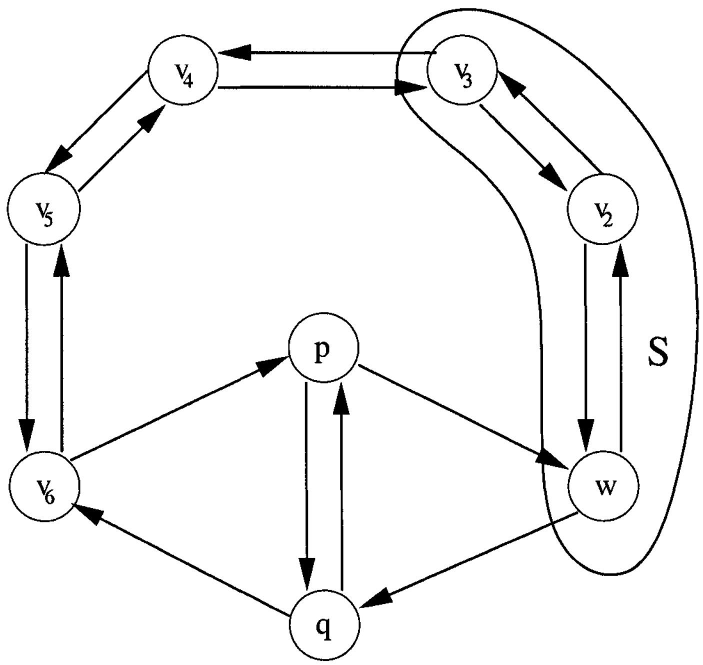

Figure 6.4: x when $k = 3$ and $n = 8$

Clearly, x is a feasible point of the ASEP since it can be obtained by repeatedly applying the 2-jack operation to the unique (up to isomorphism) half-integer extreme point of the ASEP on 4 nodes. It is straightforward to compute that $\alpha x = -\frac{1}{2}$.

Now consider a dual solution given by

$$(y_v^{\text{in}}, y_v^{\text{out}}) = \begin{cases} (0, -\frac{1}{2}) & \text{if } v = p \\ (-\frac{1}{2}, 0) & \text{if } v = q \text{ or } v = w \\ (0, 0) & \text{otherwise} \end{cases}$$

and

$$d_T = \begin{cases} \frac{1}{2} & \text{if } T = S \text{ or } T = S\backslash\{w\} \\ 0 & \text{otherwise.} \end{cases}$$

Next, let $uv \in E$. If $uv \in \delta(S)\backslash\{wq\}$ then

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset T \subset V} (d_T \mid uv \in \delta(T)) \leq 0 + 0 + 1 = \alpha_{uv}.$$

If $uv \in \delta(S\backslash\{w\})$ but $uv \notin \delta(S)$ then $u \in S\backslash\{w\}$ and $v = w$. Hence

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset T \subset V} (d_T \mid uv \in \delta(T)) = 0 - \frac{1}{2} + \frac{1}{2} = 0 = \alpha_{uv}.$$

In all other cases we know that

$$\sum_{\emptyset \subset T \subset V} (d_T \mid uv \in \delta(T)) = 0$$

and so

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset T \subset V} (d_T \mid uv \in \delta(T)) \leq 0.$$

If $uv = pw$ then

$$y_p^{\text{out}} + y_w^{\text{in}} + \sum_{\emptyset \subset T \subset V} (d_T \mid pw \in \delta(T)) = -\frac{1}{2} - \frac{1}{2} + 0 = -1 = \alpha_{pw}.$$

If $uv = pq$ then

$$y_p^{\text{out}} + y_q^{\text{in}} + \sum_{\emptyset \subset T \subset V} (d_T \mid pq \in \delta(T)) = -\frac{1}{2} - \frac{1}{2} + 0 = -1 = \alpha_{pq}.$$

Lastly, if $uv = wq$ then

$$y_w^{\text{out}} + y_q^{\text{in}} + \sum_{\emptyset \subset T \subset V} (d_T \mid wq \in \delta(T)) = 0 - \frac{1}{2} + \frac{1}{2} = 0 = \alpha_{wq}.$$

Thus, our dual solution is in fact feasible.

The objective value of this dual is

$$\sum_{v \in V} y_v^{\text{out}} + \sum_{v \in V} y_v^{\text{in}} + \sum_{\emptyset \subset T \subset V} d_T = -\frac{1}{2}.$$

Since this value is the same as αx we know that x is an optimal solution with respect to α. Therefore,

$$\text{ATSP}_{LP}(\alpha) = -\frac{1}{2}.$$

□

Corollary 6.5.2. *For each k*

$$Str(F) = \frac{6}{5}.$$

Proof. As noted above, $\sum_{v \in V} \alpha_{vv} = 3$, and by Theorem 6.5.1 we have that $\text{ATSP}_{LP}(\alpha) = -\frac{1}{2}$. Thus,

$$\begin{aligned} \text{Str}(F) &= \frac{0+3}{-\frac{1}{2}+3} \\ &= \frac{6}{5} \end{aligned}$$

□

In the next section, we examine the $C2$ inequalities which are similar to the T_k inequalities.

6.6 $C2$ Inequalities

In the same way that the T_k inequalities are slight modifications of cut constraints, the $C2$ inequalities are modified comb inequalities (clique tree inequalities with just one handle). As such, let $\emptyset \subset H \subset V$ and let $T_1, \ldots, T_s$ be disjoint subsets of V such that

- $s \geq 3$ is odd,
- $T_i \cap H \neq \emptyset$ for each $1 \leq i \leq s$, and
- $T_i - H \neq \emptyset$ for each $1 \leq i \leq s$.

Lastly, let p and q be distinct nodes of V which are not in H nor in T_i for any $1 \leq i \leq s$. The $C2$ inequality is

$$-x(\gamma(H)) - \sum_{i=1}^{s} x(\gamma(T_i)) - \sum_{v \in H}(x_{pv} + x_{vq}) - x_{pq} \geq -|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-1}{2}.$$

We use $\alpha x \geq \alpha_0$ to denote the $C2$ inequality.

Let $\tilde{H} = H - \cup_{i=1}^{s} T_i$ and let $\tilde{V} = V \backslash (\cup_{i=1}^{s} T_i \cup H \cup \{p, q\})$. We now compute α_{ww} for each $w \in V$.

Proposition 6.6.1.

$$\alpha_{ww} = \begin{cases} 1 & \textit{if } w = p \textit{ or } w = q \\ 1 & \textit{if } w \in \tilde{H} \\ 1 & \textit{if } w \in T_i - H \textit{ for some } 1 \leq i \leq s \\ 2 & \textit{if } w \in H \cap T_i \textit{ for some } 1 \leq i \leq s \\ 0 & \textit{otherwise} \end{cases}$$

and so

$$\sum_{w \in V} \alpha_{ww} = |H| + \sum_{i=1}^{s} |T_i| + 2.$$

Proof. We consider six cases.

Case 1: $w = p$.

Then the only arcs incident to p with non-zero α-values have their heads in $H \cup \{q\}$ and their α-value is -1. Hence we can have at most one such arc. If we choose $v = q$ and any $u \in V \backslash \{p, q\}$ we get

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 1$$

and the result follows.

Case 2: $w = q$.

Then the only arcs incident to q with non-zero α-values have their tails in $H \cup \{p\}$ and their α-value is -1. Hence we can have at most one such arc. If we choose $u = p$ and any $v \in V \backslash \{p, q\}$ we get

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 1$$

and the result follows.

Case 3: $w \in \tilde{H}$.

The only arcs incident to w with non-zero α-values have their tails in $H \cup \{p\}$ or their heads in $H \cup \{q\}$ and their α-value is -1. However, if we choose $u \in H \cup \{p\}$ and $v \in H \cup \{q\}$ then $\alpha_{uv} \leq -1$. If we choose $u = p$ and $v = q$ we get

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 1$$

and the result follows.

Case 4: $w \in T_i - H$ for some $1 \leq i \leq s$.

The only arcs incident to w with non-zero α-values have their other endpoint in T_i and their α-value is -1. If $u, v \in T_i$ then $\alpha_{uv} \leq -1$. If we choose $u \in H \cap T_i$ and $v = p$ say then

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 1$$

and the result follows.

Case 5: $w \in H \cap T_i$ for some $1 \leq i \leq s$.

Then only arcs incident to w with α-values of -2 have their other endpoint in $H \cap T_i$. If we choose $u \in H \cap T_i$, say, then $\alpha_{uv} = \alpha_{wv}$ for any $v \in V \backslash \{u, w\}$. Hence

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 2.$$

The same is true if we choose $v \in H \cap T_i$. However, it could be that $|H \cap T_i| = 1$. Conversely, if we choose $u, v \notin H \cap T_i$ then $\alpha_{uw} \geq -1$ and $\alpha_{wv} \geq -1$. Specifically, if we choose $u \in T_i - H$ and $v \in H - T_i$ then we have

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 2$$

and the result follows.

Case 6: $w \in \tilde{V}$.

All arcs incident to w have and α-value of 0. Hence, if we choose $u = q$ and $v = p$ we get

$$\alpha_{uv} - \alpha_{uw} - \alpha_{wv} = 0$$

and the result follows.

Since each node of H and each node of T_i for each $1 \leq i \leq s$ can be thought of as separately contributing 1 to the sum $\sum_{w \in V} \alpha_{ww}$ and p and q each contribute 1 we get

$$\sum_{w \in V} \alpha_{ww} = |H| + \sum_{i=1}^{s} |T_i| + 2.$$

□

In our next proof, we present a feasible solution to the ASEP and calculate its optimal value with respect to α. Again, this point can be obtained from repeatedly applying the 2-jack operation to the unique half-integer extreme point of the ASEP on 4 nodes so we know that the point is feasible.

In order to define our feasible point, we let $v_1, v_2, \ldots, v_{n-2}$ be the nodes of $V \backslash \{p, q\}$ ordered as follows:

1) The nodes of $\tilde{H}$ followed by

2) the nodes of $T_1 \cap H$ followed by the nodes of $T_1 - H$ followed by

3) the nodes of $T_2 - H$ followed by the nodes of $T_2 \cap H$ followed by

4) the nodes of $T_3 \cap H$ followed by the nodes of $T_3 - H$ followed by

5) the nodes of $T_4 - H$ followed by the nodes of $T_4 \cap H$ followed by

$\vdots$

s) the nodes of $T_{s-1} - H$ followed by the nodes of $T_{s-1} \cap H$ followed by

s+1) the nodes of $T_s \cap H$ followed by the nodes of $T_s - H$ followed by

s+2) the nodes of $\tilde{V}$.

Theorem 6.6.2.

$$ATSP_{LP}(\alpha) = -|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-2}{2}.$$

Proof. We define x as follows

$$x_{uv} = \begin{cases} \frac{1}{2} & \text{if } uv = v_i v_{i+1} \text{ or } uv = v_{i+1} v_i \text{ for some } 1 \leq i \leq n-3 \\ & \text{or } uv \in \{pv_1, v_1 q, pq, qp, v_{n-2} p, q v_{n-2}\} \\ 0 & \text{otherwise.} \end{cases}$$

Notice that each tooth T_i is a tight set of x and so $x(\gamma(T_i)) = |T_i| - 1$ for each $1 \leq i \leq s$. Furthermore, there are precisely $s+1$ arcs of $\delta(H)$ in the support digraph of x so $x(\gamma(H)) = |H| - \frac{s+1}{2}$. Thus

$$\begin{aligned} \alpha x &= -(|H| - \frac{s+1}{2}) - \sum_{i=1}^{s}(|T_i| - 1) - \frac{3}{2} \\ &= -|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-2}{2}. \end{aligned}$$

Now, in his proof of validity of the C2 inequality for the ATSP-polytope, Grötschel [43] shows that

$$-x(\gamma(H)) - \sum_{i=1}^{s} x(\gamma(T_i)) - \sum_{v \in H}(x_{pv} + x_{vq}) - x_{pq} \geq -|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-2}{2}$$

was a valid inequality for the ASEP.

Therefore our solution is in fact optimal and so

$$\text{ATSP}_{LP}(\alpha) = -|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-2}{2}.$$

□

We now have all the pieces we need to calculate the strength of the $C2$ inequality.

Theorem 6.6.3. *Let F be the facet induced by the $C2$ inequality.*

$$\mathit{Str}(F) = \frac{3s+3}{3s+2}$$

and so for all $s \geq 3$,

$$\mathit{Str}(F) \leq \frac{12}{11}.$$

Proof. Since

$$\sum_{v \in V} \alpha_{vv} = |H| + \sum_{i=1}^{s} |T_i| + 2,$$

$$\text{ATSP}(\alpha) = -|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-1}{2},$$

and

$$\text{ATSP}_{LP}(\alpha) = -|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-2}{2},$$

we have that

$$\begin{aligned}
\text{Str}(F) &= \frac{-|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-1}{2} + |H| + \sum_{i=1}^{s} |T_i| + 2}{-|H| - \sum_{i=1}^{s}(|T_i| - 1) + \frac{s-2}{2} + |H| + \sum_{i=1}^{s} |T_i| + 2} \\
&= \frac{s + \frac{s-1}{2} + 2}{s + \frac{s-2}{2} + 2} \\
&= \frac{\frac{3s+3}{2}}{\frac{3s+2}{2}} \\
&= \frac{3s+3}{3s+2}.
\end{aligned}$$

However, since $s \geq 3$ we get that

$$\begin{aligned}
\text{Str}(F) &= 1 + \frac{1}{3s+2} \\
&\leq \frac{12}{11}
\end{aligned}$$

and we see that the maximum strength is attained when $s = 3$. □

We turn our attention to the last facet-inducing inequality we examine – the $C3$ inequalities.

6.7 $C3$ Inequalities

Let T and U be disjoint subsets of V such that $|T|, |U| \geq 2$. Let t_1 be a fixed node of T and let u_1 be a fixed node of U. Lastly, let $W = V \backslash (T \cup U)$ and let w_1 be a fixed node of W. The $C3$ inequality is

$$-x(\gamma(T)) - x(\gamma(U)) - x_{u_1 t_1} - x_{w_1 t_1} - x_{w_1 u_1} - \sum_{v \in U} x_{t_1 v} \geq -|T| - |U| + 1.$$

This is the negative of the more common presentation of this inequality so that the inequality sign points in the proper direction. Let $\alpha x \geq -|T| - |U| + 1$ denote the $C3$ inequality.

Figure 6.5 depicts the $C3$ inequality where $q = 2$, $r = 2$, and $s = 3$. Each arc shown has an α-value of -1.

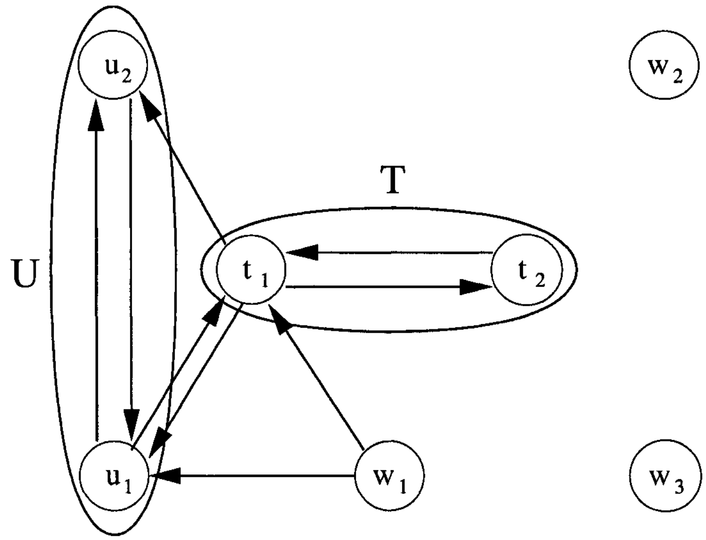

Figure 6.5: The $C3$ inequality with $q = 2$, $r = 2$, and $s = 3$

It is not too difficult to compute

$$\alpha_{vv} = \begin{cases} 2 & \text{if } v = t_1 \\ 2 & \text{if } v = u_1 \\ 1 & \text{if } v = w_1 \\ 1 & \text{if } v \in T \backslash \{t_1\} \\ 1 & \text{if } v \in U \backslash \{u_1\} \\ 0 & \text{otherwise} \end{cases}$$

and so

$$\sum_{v \in V} \alpha_{vv} = |T| + |U| + 3.$$

Again, we present a feasible ASEP point and prove that it is an optimal solution of the ATSP_{LP} with respect to α by presenting a dual solution with the same objective value. To facilitate this, we label the nodes of T, U, and W. Namely, let $t_2, \ldots, t_q$ be the nodes of $T\backslash\{t_1\}$, let $u_2, \ldots, u_r$ be the nodes of $U\backslash\{u_1\}$, and let $w_1, \ldots, w_s$ be the nodes of $W\backslash\{w_1\}$.

Theorem 6.7.1.

$$ATSP_{LP}(\alpha) = -|T| - |U| + \frac{1}{3}.$$

Proof. We now define a feasible point x of the ASEP. Let

$$x_{uv} = \begin{cases} \frac{2}{3} & \text{if } uv \in \{w_1u_1, u_rw_2, w_sw_1\} \\ & \text{or } uv = t_it_{i+1} \text{ for some } 1 \leq i \leq q-1 \\ & \text{or } uv = u_iu_{i+1} \text{ for some } 1 \leq i \leq r-1 \\ & \text{or } uv = w_iw_{i+1} \text{ for some } 2 \leq i \leq s-1 \\ \frac{1}{3} & \text{if } uv \in \{t_qw_1, w_1t_1, u_1t_1, t_1u_r, w_2t_q, t_qw_s\} \\ & \text{or } uv = t_{i+1}t_i \text{ for some } 1 \leq i \leq q-1 \\ & \text{or } uv = u_{i+1}u_i \text{ for some } 1 \leq i \leq r-1 \\ & \text{or } uv = w_{i+1}w_i \text{ for some } 2 \leq i \leq s-1 \\ 0 & \text{otherwise.} \end{cases}$$

A picture of x is shown in Figure 6.6 for $q = 2$, $r = 2$, and $s = 3$. The solid arcs have x-values of $\frac{2}{3}$ whereas the dashed lines have an x-value of $\frac{1}{3}$.

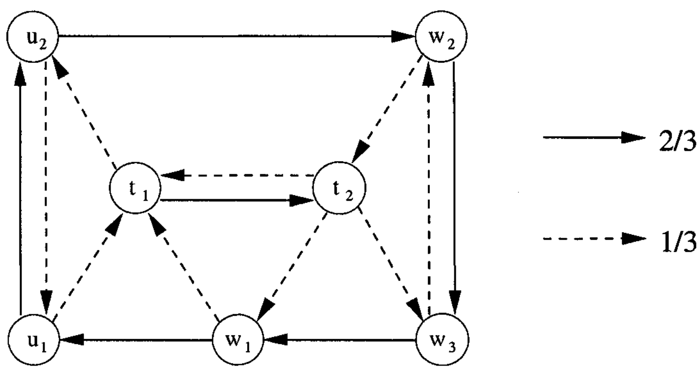

Figure 6.6: x with $q = 2$, $r = 2$, and $s = 3$

To see that x is in fact feasible, it is easy to check that the node equalities hold at every node. Notice that the arcs of x of value $\frac{1}{3}$ consist of the dicycles

$(u_r, u_{r-1}, \ldots, u_1, t_1)$ and $(w_s, w_{s-1}, \ldots, w_2, t_q)$ and the (t_q, t_1)-dipaths (t_q, w_1, t_1) and $(t_q, t_{q-1}, \ldots, t_1)$. Thus, if $S \subset V$ is such that $\delta(S)$ contains no arc of value $\frac{1}{3}$ then either $S = U \cup \{t_1, \ldots, t_i\}$ for some $1 \leq i \leq q-1$ or $S = U \cup \{w_1, t_1, \ldots, t_i\}$ for some $1 \leq i \leq q-1$. In either case, it must be that $u_r w_2, t_i t_{i+1} \in \delta(S)$ and these are both arcs of value $\frac{2}{3}$. Hence $\delta(S) \geq \frac{4}{3}$.

As for the arcs of x of value $\frac{2}{3}$, they form the dicycle $(u_1, u_2, \ldots, u_r, w_2, w_3, \ldots, w_s, w_1)$ and the (t_1, t_q)-dipath $(t_1, t_2, \ldots, t_q)$. Hence, if $S \subset V$ is such that $\delta(S)$ contains no arc of value $\frac{2}{3}$ then either $S = \{t_i, \ldots, t_q\}$ for some $1 \leq i \leq q$ or $S = V \backslash \{t_1, \ldots, t_i\}$ for some $1 \leq i \leq q$. In the former case, $\delta(S)$ must contain the arcs $t_q w_1$, $t_q w_s$, and either $t_i t_{i-1}$ or $t_1 u_r$. In the latter case, $\delta(S)$ must contain the arcs $w_1 t_1$, $u_1 t_1$, and either $t_{i+1} t_i$ or $w_2 t_q$. Thus, in all cases, $\delta(S)$ contains at least 3 arcs of value $\frac{1}{3}$ and so $\delta(S) \geq 1$.

For all other $\emptyset \subset S \subset V$, $\delta(S)$ must contain at least one arc of value $\frac{1}{3}$ and at least one arc of value $\frac{2}{3}$. Therefore, $\delta(S) \geq 1$ and so x is indeed a feasible point of the ASEP.

Now let us introduce the dual variables. Let

$$(y_v^{\text{in}}, y_v^{\text{out}}) = \begin{cases} (-\frac{2}{3}, -1) & \text{if } v = t_1 \\ (-\frac{2}{3}, -\frac{2}{3}) & \text{if } v = u_1 \\ (0, -\frac{1}{3}) & \text{if } v = w_1 \\ (0, -1) & \text{if } v \in T \backslash \{t_1\} \\ (-\frac{1}{3}, -\frac{2}{3}) & \text{if } v \in U \backslash \{u_1\} \\ (0, 0) & \text{otherwise} \end{cases}$$

and let

$$d_S = \begin{cases} \frac{2}{3} & \text{if } S = T \backslash \{t_1\} \\ \frac{1}{3} & \text{if } S = T,\ S = U, \text{ or } S = U \backslash \{u_1\} \\ 0 & \text{otherwise.} \end{cases}$$

To prove that our dual solution is in fact feasible, notice first that $y_v^{\text{in}} \leq 0$ and $y_v^{\text{out}} \leq 0$ for all $v \in V$. Let $uv \in E$. If $u \in T$ then $y_u^{\text{out}} = -1$ and

$\sum_{\emptyset\subset S\subset V}(d_S \mid uv \in \delta(S)) \leq 1$. If $u \in U$ then $y_u^{\text{out}} = -\frac{2}{3}$ and $\sum_{\emptyset\subset S\subset V}(d_S \mid uv \in \delta(S)) \leq \frac{2}{3}$. Otherwise we have $\sum_{\emptyset\subset S\subset V}(d_S \mid uv \in \delta(S)) = 0$. Therefore, in all cases,

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid uv \in \delta(S)) \leq 0.$$

All that remains is to check the feasibility constraint for $uv \in E$ such that $\alpha_{uv} < 0$. To do this, we consider 9 cases. For each case, $\alpha_{uv} = -1$.

Case 1: $u \in T$ and $v \in T\backslash\{t_1\}$.

$$\begin{aligned} y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid uv \in \delta(S)) &= -1+0+0 \\ &= -1. \end{aligned}$$

Case 2: $u \in U$ and $v \in U\backslash\{u_1\}$.

$$\begin{aligned} y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid uv \in \delta(S)) &= -\frac{2}{3}-\frac{1}{3}+0 \\ &= -1. \end{aligned}$$

Case 3: $u \in T\backslash\{t_1\}$ and $v = t_1$.

$$\begin{aligned} y_u^{\text{out}} + y_{t_1}^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid ut_1 \in \delta(S)) &= -1-\frac{2}{3}+\frac{2}{3} \\ &= -1. \end{aligned}$$

Case 4: $u \in U\backslash\{u_1\}$ and $v = u_1$.

$$\begin{aligned} y_u^{\text{out}} + y_{u_1}^{\text{in}} + \sum_{\emptyset\subset S\subset V}(d_S \mid uu_1 \in \delta(S)) &= -\frac{2}{3}-\frac{2}{3}+\frac{1}{3} \\ &= -1. \end{aligned}$$

Case 5: $u = t_1$ and $v \in U\backslash\{u_1\}$.

$$\begin{aligned} y_{t_1}^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid t_1 v \in \delta(S)) &= -1 - \frac{1}{3} + \frac{1}{3} \\ &= -1. \end{aligned}$$

Case 6: $uv = t_1 u_1$.

$$\begin{aligned} y_{t_1}^{\text{out}} + y_{u_1}^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid t_1 u_1 \in \delta(S)) &= -1 - \frac{2}{3} + \frac{1}{3} \\ &= -\frac{4}{3}. \end{aligned}$$

Case 7: $uv = u_1 t_1$.

$$\begin{aligned} y_{u_1}^{\text{out}} + y_{t_1}^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid u_1 t_1 \in \delta(S)) &= -\frac{2}{3} - \frac{2}{3} + \frac{1}{3} \\ &= -1. \end{aligned}$$

Case 8: $uv = w_1 t_1$.

$$\begin{aligned} y_{w_1}^{\text{out}} + y_{t_1}^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid w_1 t_1 \in \delta(S)) &= -\frac{1}{3} - \frac{2}{3} + 0 \\ &= -1. \end{aligned}$$

Case 9: $uv = w_1 u_1$.

$$\begin{aligned} y_{w_1}^{\text{out}} + y_{u_1}^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid w_1 u_1 \in \delta(S)) &= -\frac{1}{3} - \frac{2}{3} + 0 \\ &= -1. \end{aligned}$$

Thus we see that our dual solution is indeed feasible. The objective value is

$$\begin{aligned} \sum_{v \in V} y_v^{\text{out}} + \sum_{v \in V} y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} d_S &= (-|T| - \frac{2}{3}|U| - \frac{1}{3}) + (-\frac{2}{3} - \frac{1}{3}|U| - \frac{1}{3}) + \frac{5}{3} \\ &= -|T| - |U| + \frac{1}{3}. \end{aligned}$$

By looking at the above cases and some simple other cases, we can see that whenever $x_{uv} > 0$ we have that

$$y_u^{\text{out}} + y_v^{\text{in}} + \sum_{\emptyset \subset S \subset V} (d_S \mid uv \in \delta(S)) = \alpha_{uv}.$$

Additionally, if $d_S > 0$ then $x(\delta(S)) = 1$. Hence by the complementary slackness conditions, we have that x is an optimal solution with respect to α. Therefore,

$$\text{ATSP}_{LP}(\alpha) = -|T| - |U| + \frac{1}{3}.$$

□

We now use the result from Theorem 6.7.1 to compute the strength of the $C3$ inequality.

Corollary 6.7.2. *Let F be the facet induced by the $C3$ inequality. Then,*

$$\mathit{Str}(F) = \frac{6}{5}.$$

Proof. $\text{ATSP}(\alpha) = -|T| - |U| + 1$, $\text{ATSP}_{LP}(\alpha) = -|T| - |U| + \frac{1}{3}$, and $\sum_{v \in V} \alpha_{vv} = |T| + |U| + 3$. Thus,

$$\begin{aligned} \text{Str}(F) &= \frac{(-|T| - |U| + 1) + (|T| + |U| + 3)}{(-|T| - |U| + \frac{1}{3}) + (|T| + |U| + 3)} \\ &= \frac{4}{\frac{10}{3}} \\ &= \frac{6}{5}. \end{aligned}$$

□

6.8 Summary

Here we collect the results from this section regarding the strength of the various classes of facet-inducing inequalities. For each facet-inducing inequality, we present the maximum strength for the resulting facets. The final column states when this maximum strength is attained.

Facet-inducing Inequality	Maximum Strength	Occurrence
Clique Tree	8/7	$h_i = 3$ for all $1 \leq i \leq r$, $t_j \leq 2$ for all $1 \leq j \leq s$, and $r \to \infty$
D_k^-	6/5	$k = 3$ or $k = 4$
D_k^+	6/5	$k = 3$ or $k = 4$
T_k	6/5	always
$C2$	12/11	$s = 3$
$C3$	6/5	always

Hence we can see, among all the facets studied, that the greatest strength is $\frac{6}{5}$. However, we know that the strength of a facet is also the integrality gap of a certain set of metric arc costs. In the last chapter, we see that the integrality gap is at least 2. This contrasts sharply from the best integrality gap we find in this chapter by computing the strength of the facets. In the next chapter, we turn our attention back to the ASEP and discover some interesting numerical results about its extreme points. We also present a new heuristic for generating many of the non-isomorphic extreme points of the ASEP.

Chapter 7

Unique Cobases of ASEP Extreme Points

We present in Chapter 3 a method for generating the non-isomorphic extreme points of the ASEP for each $4 \leq n \leq 7$. We also describe how we generate all the non-isomorphic half-integer extreme points for $n = 8$ and $n = 9$. Furthermore, we use the 1-arc insertion and 2-jack operations to generate other half-integer extreme points for even larger values of n. For $4 \leq n \leq 7$ we use PORTA and then CDD+ to generate all the extreme points and then use NAUTY to output a single representative of each isomorphism class of points. The reason that we could not extend this method to find all non-isomorphic extreme points for $n = 8$ is that these programs would take an impractical amount of time. We conjecture that the reason for this is that there are so many isomorphic extreme points and each of these programs generate many of them. We are interested in developing an algorithm for generating extreme points of the ASEP that would take into account this symmetry and hopefully drastically reduce the workload.

In this chapter, we present an algorithm for generating the extreme points of the ASEP which takes isomorphic symmetry into account. The underlying method used in PORTA and CDD+ is the Double Description Method [67]. In the Double Description method, we find the extreme points of a polyhedron by sequentially adding the constraints defining the polyhedron. At each step, we know all the extreme points of the polyhedron defined by the current set of constraints. When we add a new constraint to get a new polyhedron, we remove any extreme points of the old polyhedron which do not satisfy the new constraint and we add any new extreme points we find which lie on the hyperplane corresponding to the most recently added constraint. In this way, we eventually add all the constraints and hence have a list of the extreme points of the polyhedron.

However, we do not use the double description method. Instead, we use a pivoting method (as did Avis and Fukuda [3] in the pivoting algorithm lrs) based on the Simplex Method. However, such a pivoting method can be very inefficient if the extreme points of the polytope have many different cobases (we call such extreme points *degenerate*). We explain in the next section how we overcome this hurdle by only choosing to pivot at extreme points of the ASEP which have very few cobases. Unfortunately, we do not find all the extreme points of the ASEP using this method but we see in the following sections that we can use this method to get many of the non-isomorphic extreme points of the ASEP in a more reasonable amount of time.

7.1 Unique Cobases

In this section we introduce the idea that an ASEP extreme point could be restricted to have very few cobases. We see that we can never have a unique cobasis in the strictest mathematical definition of the word, but it turns out that there are certain extreme points which have cobases which are unique up to certain trivial requirements. The benefit of finding extreme points with these "unique" cobases is that we can avoid degeneracy in a pivoting algorithm (such as the Simplex Method) when trying to map out the parts of the ASEP near these given extreme points.

Balinski and Russakoff [5] remark in their paper "On the Assignment Polytope" that the node equalities of the assignment polytope (which are also in the description of the ASEP) are linearly dependent but the rank of the node equalities is precisely $2n-1$. This linear dependence is easy to see since

$$\sum_{v \in V} x(\delta(V)) = \sum_{uv \in E} x_{uv} = \sum_{v \in V} x(\delta(V \backslash \{v\})).$$

Hence we can build up a cobasis for any ASEP extreme point from all but one of the node equalities. Furthermore, any of the node equalities can be the one left out of the cobasis, so already, for every ASEP extreme point, we have $2n$ different choices of linearly independent sets of constraints that we can build up to a cobasis. However, acknowledging this linear dependence, we can fix some node z and we can adopt the convention that the cobasis we build for each extreme point of the ASEP contains all the node equalities except, say, $x(\delta(V\backslash\{z\})) = 1$. Let's call this convention *C1.*

Secondly, we observe in Proposition 3.2.1 that if $\delta(S)$ is a tight cut of an extreme point x then so is $\delta(V\backslash S)$. More importantly, we see in the proof of this proposition that

$$x(\delta(V \backslash S)) = x(\delta(S)) - \sum_{v \in S} x(\delta(v)) + \sum_{v \in S} x(\delta(V \backslash \{v\})).$$

Hence, no cobasis of x can contain all the node equalities (or all but one of them since the missing one is implied), $x(\delta(S)) = 1$, and $x(\delta(V\backslash S)) = 1$. Again, we can acknowledge this linear dependence and adopt the convention that the cobasis of an extreme point only includes a tight cut constraint $x(\delta(S)) = 1$ if $|S| < n/2$ or $|S| = n/2$ and $z \in S$. Hence, by applying this convention, we uniquely specify among the two complements which tight set induces a cut constraint that we include in the cobasis. We call this convention *C2.*

Consider the tour on 3 nodes shown in Figure 7.1. This is an extreme point of the ASEP on 3 nodes and it has 50 different cobases. Furthermore, 15 of these cobases obey convention C1. For the case $n = 3$, convention C2 is not applicable. For a more interesting example, consider the half-integer extreme point shown in Figure 7.2. Depending on our choice of z, there are either 12 or 14 cobases that satisfy conventions C1 and C2. Hence we see that an ASEP extreme point can have

n	Extreme Points	Unique Cobasis
3	1	0
4	2	1
5	5	1
6	90	59
7	3748	2694

Table 7.1: Number of extreme points with "unique" cobases

many cobases. However, the extreme point shown in Figure 7.3 (where dashed lines depict arcs with x-values of $1/3$ and solid lines depict arcs with x-values of $2/3$) has a unique cobasis that satisfies conventions C1 and C2. Hence we can also have extreme points of the ASEP which have unique cobases in this sense. Table 7.1 details how common it is for an extreme point to have a unique cobasis which satisfies C1 and C2. The second column contains the number of non-isomorphic extreme points for each value of n. The third column of the table displays the number of non-isomorphic extreme points for each value of n which have a unique cobasis which obeys convention C1 and C2.

In the next section, we present an algorithm similar to the Simplex Method, for generating extreme points of the ASEP. We can avoid degeneracy by only pivoting at extreme points that have unique cobases which obey conventions C1 and C2.

7.2 A New Algorithm for Generating ASEP Extreme Points

In this section, we use the concept of an extreme point having a unique cobasis with respect to C1 and C2 and perform a depth first search to generate many (if not all) of the non-isomorphic extreme points of the ASEP. The input of this program is an

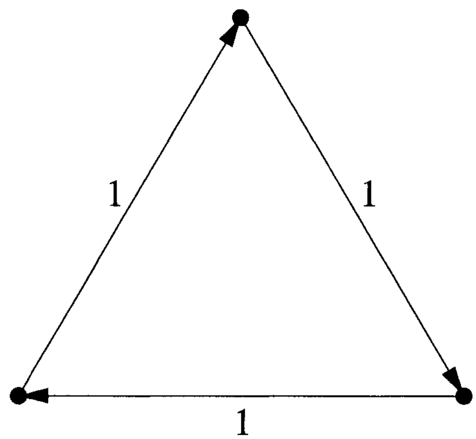

Figure 7.1: The tour on 3 nodes

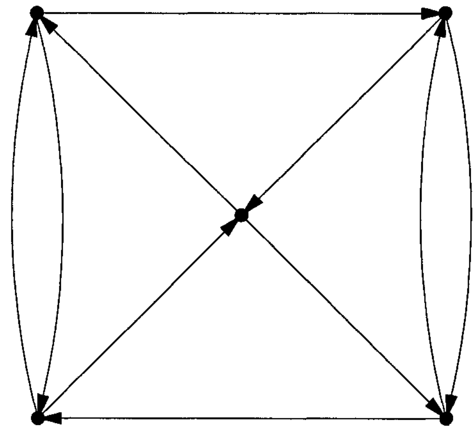

Figure 7.2: A half-integer extreme point with many cobases

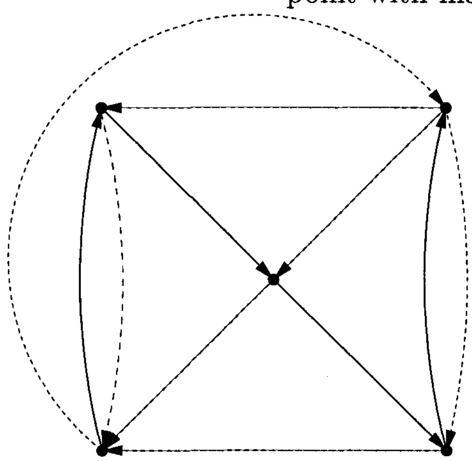

Figure 7.3: An extreme point with a "unique" cobasis

extreme point of the ASEP-polytope with a unique cobasis which obeys conventions C1 and C2. The output is a list of non-isomorphic extreme points of the ASEP along with their adjacency relationships on the polytope. Again, we use NAUTY to recognize when we have generated an isomorphic copy of some extreme point already in the list. Since our algorithm is based on the Simplex Method, we only pivot on extreme points with unique cobases which obey C1 and C2 to eliminate degeneracy.

Our first step in the recursive algorithm is to rewrite the constraints of the

ASEP-polytope in equality form with slack variables. We then construct the cobasis matrix and, using Gaussian elimination, compute the inverse. Now consider a generic extreme point, x, of the ASEP-polytope with a unique cobasis which obeys C1 and C2 and let B_x be the submatrix of the constraint matrix whose columns are indexed by the basic variables of x. Here, we think of x both containing the slack of each nonnegativity constraint and the slack for each cut constraint. Let N be the set of indices of the non-basic variables of x. Furthermore, let A_k be the column of the constraint matrix corresponding to the variable x_k. Lastly, let $\mathcal{L}$ be the list of non-isomorphic extreme points generated so far and let $\mathcal{N}(x)$ denote the list of known neighbours of x (extreme points of the ASEP which can be obtained from a single pivot starting from x) on the ASEP for each $x \in \mathcal{L}$ with $\mathcal{N}$ denoting the set of all $\mathcal{N}(x)$ for $x \in \mathcal{L}$. If we find that there are several distinct neighbours of x which are mutually isomorphic then we also keep track of the number of times a neighbour from such an equivalence class appears as a neighbour of x. The main function in our program is the depth-first search described in the box on the following page.

In essence this algorithm is the Simplex Method. During Step 5 we can keep track of which basic variables have been set to zero. If there is more than one such variable, we know that y cannot have a unique cobasis with respect to C1 and C2 since we only have cut constraints which obey convention C2. In Step 6 we compute our canonicalization of y (recall that y is an extreme point of the ASEP along with extra information telling us about the slack on each of the cut constraints) using NAUTY just as we do in Chapter 4.

A paper by McKay [62] discusses a very beautiful and general framework for generating non-isomorphic combinatorial objects. He describes how we can check whether a given labelled object generated in the depth first search tree used to generate non-isomorphic combinatorial objects is the canonical representative for its equivalence class. The knowledge of such a canonical representative would eliminate the need for us to store and reference the list $\mathcal{L}$ and we could simply use his canonicity test

procedure `DepthFirstSearch` $(x, B_x^{-1}, N, \mathcal{L}, \mathcal{N})$

1. **If** N is empty **then**

   return.

2. Choose a non-basic index, $k \in N$, and remove k from N.

3. Compute $d = B_x^{-1} A_k$.

4. Compute $t = \min\{\frac{x_i}{d_i} \mid x_i > 0,\ d_i > 0\}$.

5. Compute $y_i = \begin{cases} x_i - td_i & \text{if } i \text{ is basic.} \\ t & \text{if } i = k. \\ x_i & \text{otherwise.} \end{cases}$

6. Compute the canonicalization, y', of y.

7. Add y' to $\mathcal{N}(x)$.

8. **If** y' is not in our list $\mathcal{L}$ **then**

   Add y' to $\mathcal{L}$.

   If y has a unique cobasis **then**

   Create N', the set of non-basic indices of y.

   Compute B_y^{-1} from B_x^{-1}.

   Call `DepthFirstSearch` $(y, B_y^{-1}, N', \mathcal{L}, \mathcal{N})$.

   Recover B_x^{-1} from B_y^{-1}.

9. Call `DepthFirstSearch` $(x, B_x^{-1}, N, \mathcal{L}, \mathcal{N})$.

to determine whether or not to output the given extreme point. Unfortunately, his framework requires an invertible recursive definition of the combinatorial objects. In our case, we do not know of a recursive way of creating all the extreme points of the ASEP on n nodes from the extreme points of the ASEP on fewer than n nodes (although this idea provides some motivation for our exploration of the operations we present in Chapter 3).

Avis and Fukuda's [3] describe a way of generating all the extreme points of a polyhedron using a pivoting algorithm. They also avoid the use of a running list of extreme points (what would be equivalent to $\mathcal{L}$ in our algorithm) by first finding all the cobases of all the extreme points of the polyhedron and defining for all but one cobasis a unique cobasis parent. They then introduce the ideea of a canonical cobasis (via a lexicographical ordering rule) for each extreme point to avoid outputting the same extreme point twice. As a result, they do not need to store a list of the extreme points generated so far. For our algorithm, however, we do not generate every possible cobasis of every single extreme point of the ASEP. In fact, we do not explore the cobases of degenerate extreme points nor do we even generate all the extreme points of the polytope. It is unclear whether the ideas of Avis and Fukuda can be introduced into our algorithm to eliminate the need for the list $\mathcal{L}$.

In Step 8, we do not want to recompute the inverse cobasis matrix from scratch each time we generate a new extreme point. Since there is a unique index, r, of the variable leaving the cobasis, B_y^{-1} can be obtained from B_x^{-1} by replacing the column corresponding to r with A_k. In this case, B_y^{-1} can be computed as follows.

$$(B_y^{-1})_{ij} = \begin{cases} \frac{1}{d_r}(B_x^{-1})_{rj} & \text{if } i = r \\ (B_x^{-1})_{ij} - \frac{d_i}{d_r}(B_x^{-1})_{rj} & \text{otherwise.} \end{cases}$$

Similarly, we can recover B_x^{-1} from B_y^{-1} as follows.

$$(B_x^{-1})_{ij} = \begin{cases} d_k(B_y^{-1})_{kj} & \text{if } i = k \\ (B_y^{-1})_{ij} + d_i(B_y^{-1})_{kj} & \text{otherwise.} \end{cases}$$

For $n = 4$ and $n = 5$ we are able to generate all of the non-isomorphic extreme points of the ASEP using our algorithm. For $n = 6$ and $n = 7$ we are able to generate all but three of the non-isomorphic extreme points of each polytope. In either case, all three of these extreme points have a unique cobasis with respect to C1 and C2 and are surrounded on the polytope only by extreme points that do not have a unique cobasis with respect to C1 and C2. Hence, we see that the algorithm produces a large percentage, if not all, of the non-isomorphic extreme points of the polytope.

Chapter 8

Metric Costs and Optimal Strongly Connected Digraphs

In this chapter, we address the question of how much we can change the integer programming formulation of the ATSP, specifically the node equalities, and still get a lower bound on the ATSP with a constant integrality gap. We first consider the strongly connected spanning subgraph problem which is that of finding a minimum cost strongly connected directed subgraph which spans the complete digraph on n nodes with respect to some set of nonnegative arc costs. Interestingly, an integer programming formulation of this problem can be obtained from the commonly used integer programming formulation of the ATSP by simply changing "$= 1$" in all of the node equalities to "≥ 1". Hence the strongly connected spanning subgraph problem is a relaxation of the ATSP. We discover that the worst-case ratio of this problem with the ATSP is provably proportional to n, even with respect to metric cost functions. This is quite different from the undirected analogues which have a constant worst-case ratio. We also give a complete characterization of a family of optimal solutions for this relaxed problem. The important property of this family is that, when looking for an optimal solution with respect to a metric cost function, we are guaranteed to find an optimal solution within this family. Later, we see that a less drastic relaxation of the node equalities results in a new problem whose worst-case ratio with respect to the ATSP is also not constant. We finish this chapter with some computational results on the number of extreme points and rays associated with the feasible regions of linear programming relaxations of these two new problems.

8.1 Min-Cost Strongly Connected Digraphs

The *strongly connected spanning subgraph problem*, denoted the $SCSSP$, is that of finding a minimum cost strongly connected directed subgraph of the complete digraph on n nodes with respect to some set of nonnegative arc costs c. We use $SCSSP(c)$ to denote the optimal value of a given instance of this problem (and sometimes also the problem itself when the context warrants).

This problem was first studied by Moyles and Thompson [68] and Hsu [50] in the special case of finding a strongly connected spanning subgraph with a minimum number of arcs in a digraph (not necessarily a complete digraph) with uniform arc costs. These authors also provide an algorithm for solving the problem even though Garey and Johnson [37] show that the problem is NP-hard even in this special case. Later, Chopra [20] studied the case where general non-unit arc costs are considered. Monma, Munson, and Pulleyblank [66] investigate, what could be considered, the undirected analogue of the SCSSP – finding a minimum cost 2-edge-connected spanning subgraph of the complete (undirected) graph on n vertices. Furthermore, they only consider metric cost functions. They find that for every metric cost function, there is an optimal solution to the problem which belongs to a special class of 2-edge-connected graphs. In this section, we show the analogous result for the SCSSP in directed graphs. Specifically, we introduce a family of strongly connected digraphs $\mathcal{S}$ such that there is a solution to SCSSP(c) in $\mathcal{S}$ for any metric cost function c. Furthermore, we show that $\mathcal{S}$ is minimal with respect to this property – that is, each digraph in $\mathcal{S}$ is the unique optimal solution to the SCSSP with respect to some metric set of arc-costs. The reason that this family $\mathcal{S}$ is of interest is that if we want to compute the exact integrality gap for the SCSSP for small values of n in the same manner as we did for the ATSP, we could follow a similar formulation, replacing the constraints involving tours with constraints involving the digraphs in $\mathcal{S}$. This would be a great improvement over formulating a constraint for every possible strongly connected subgraph of K_n.

Let D be a strongly connected directed subgraph of the complete digraph on n nodes. We define the *canonical distance function*, d, of D on the arcs of the complete digraph as follows:

- For every arc uv in D, $d_{uv} = 1$.
- For every arc uv not in D, d_{uv} is the length of the shortest (uv)-dipath in D.

Since the canonical distance function, d, of a digraph, D, is simply the metric completion of the costs assigned to the arcs of D (which are all 1), by Lemma 4.1.1, d is metric.

Proposition 8.1.1. *If D is an arc-minimal strongly-connected directed subgraph of the complete digraph on n nodes and d is its canonical distance function, then D is a minimum-cost strongly connected directed subgraph of the complete digraph on n nodes with respect to the metric cost function d.*

Proof. Suppose D_0 is a minimum-cost strongly connected directed subgraph of the complete digraph on n nodes with respect to the metric cost function d. Replace each arc, uv, of D_0 which is not in D with a shortest (u, v)-dipath in D to obtain a multi-digraph, D'. Notice that D' is strongly connected and has the same total cost as D_0. If D' has any parallel arcs then we can remove them to get a strongly connected multi-digraph of lower cost, which contradicts the optimality of D_0. Hence D' is a digraph and thus is a directed subgraph of D. If $D' \neq D$ then D' is a proper directed subgraph of D which is strongly connected. This contradicts the arc-minimality of D. Thus $D' = D$ and since D' has the same cost as D_0, we have that D is optimal. □

We now offer the directed analogue to Monma, Munson, and Pulleyblank's [66] result. Although arc-minimal digraphs are optimal solutions to the SCSSP with respect to their own canonical distance function, we see in Proposition 8.1.2 that, when we require every node to have indegree 1 or outdegree 1, then they are the unique optimal solution. We then discover in Theorem 8.1.3 that every metric cost function has an optimal solution in this family of digraphs.

Proposition 8.1.2. *If D is an arc-minimal strongly-connected directed subgraph of the complete digraph on n nodes where every node either has outdegree exactly 1 or indegree exactly 1 and d is its canonical distance function, then D is the unique minimum cost strongly connected directed subgraph of the complete digraph on n nodes with respect to the metric cost function d.*

Proof. Let D_0 be a minimum cost strongly connected directed subgraph of the complete digraph on n nodes with respect to the metric cost function d. Suppose, for a contradiction, that uv is an arc of D_0 which is not in D and let w be an internal node of the shortest (u, v)-dipath in D which replaces uv in the construction of D' (as described in the proof of Proposition 8.1.1). Now, since D_0 is strongly connected, w must be the head of some arc of D_0 and w must also be the tail of some arc of D_0. Notice that, at each step in the construction of D', we never decrease the indegree or outdegree of any node. Thus it must be that w has outdegree at least two and indegree at least two in D'. However, w either has outdegree one or indegree one in D. But, as in the proof of Proposition 8.1.1, we have that $D' = D$. Hence no such uv can exist and so D must be the unique optimal solution. □

Theorem 8.1.3. *Let c be a metric cost function defined on the complete digraph on n nodes. Then there is a minimum-cost strongly connected digraph which is arc-minimal and where every node either has outdegree 1 or indegree 1.*

Proof. Among all of the minimum-cost strongly connected digraphs choose one, call it D, which has a minimum number of arcs. Suppose, for a contradiction, that D has a node u with outdegree at least 2 and indegree at least 2.

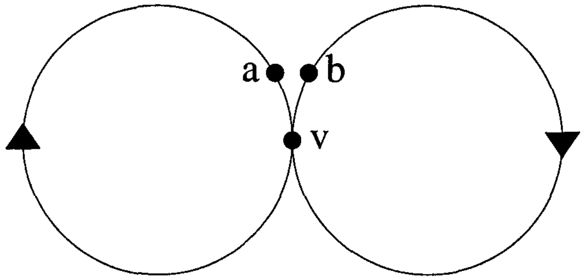

Figure 8.1: Two dicycles which intersect only at u

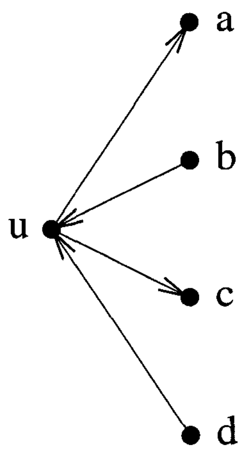

Figure 8.2: Four neighbours of u

First, we present a useful fact about D. There cannot be two dicycles in D which are node disjoint except at some node v. Otherwise, let a and b be the nodes incident to v on the distinct dicycles as shown in Figure 8.1. Notice that the digraph D', obtained by removing arcs av and vb and adding the arc ab contains both a (a, v)-dipath and a (v, b)-dipath and thus D' is strongly connected. However, since c is metric, the cost of D' is at most that of D. Thus, since D is a minimum cost strongly connected digraph, so is D'. Unfortunately, D' has fewer arcs that D, contradicting our choice of D. Therefore D does not contain two disjoint dicycles which intersect only at a single node.

Now let a, b, c, and d be distinct neighbours of u as shown in Figure 8.2.

Since D is strongly connected, it must contain an (a, b)-dipath, call it P. Notice that P cannot contain u since, otherwise, P is composed of a (a, u) dipath, call it P_1, and a (u, b) dipath, call it P_2 so that $P_1 \cup \{ua\}$ and $P_2 \cup \{bu\}$ are two dicycles of D which are node disjoint except at u and this cannot happen. Hence, P does not contain u. Similarly, let Q be a (c, d)-dipath in D. Just as with P, Q cannot conatain u. Since neither P nor Q contain u, specifically, neither P nor Q contain the arcs bu or uc.

Let D' be the digraph obtained from D by removing the arcs bu and uc and adding the arc bc (if it is not already present). Clearly, D' has fewer arcs than D, and, due to the fact that c is metric, we know that that $c_{bc} \leq c_{bu} + c_{uc}$. Hence, the cost of D' is less than or equal to the cost of D. However, since neither P nor Q contain the arcs bu or uc, $\{bc\} \cup Q \cup \{du\}$ is a walk in D' which contains a (b, u)-dipath and $\{ua\} \cup P \cup \{bc\}$ is a walk in D' which contains a (u, c)-dipath. Thus, D' is strongly connected. This contradicts the fact that D is a minimum-cost strongly connected digraph which has a minimum number of arcs. Therefore, the node u cannot exist and every node of D must either have indegree exactly one or outdegree exactly one. □

Let $\mathcal{S}$ be the set of all arc-minimally strongly connected digraphs with at least 3 nodes such that every node either has indegree 1 or outdegree 1. If c is any metric set of arc-costs of the complete directed graph on $n \geq 3$ nodes then Theorem 8.1.3 tells us that there is a solution to the min-cost strongly connected spanning subgraph problem with respect to c in $\mathcal{S}$. Hence, it is sufficient to consider the digraphs of $\mathcal{S}$ with n nodes when trying to solve the min-cost strongly connected spanning subgraph problem. Furthermore, Proposition 8.1.2 tells us that it is necessary to consider all of the digraphs in $\mathcal{S}$ with n nodes when looking for an optimal solution. Proposition 8.1.4 below describes some further properties of the digraphs in $\mathcal{S}$.

Proposition 8.1.4. *If D is an arc-minimally strongly connected digraph with at least 3 nodes such that every node either has indegree 1 or outdegree 1 then*

1. *D has no pair of antiparallel arcs;*

2. The underlying undirected support graph of D is 2-vertex-connected.

Proof. Let V be the node set of D. Suppose, for a contradiction, that D has a pair of antiparallel arcs, uv and vu. Since D has at least 3 nodes and D is strongly connected, $\delta(V\backslash\{u,v\}) \neq \emptyset$, so there are arcs, other than uv and vu, with their heads at u or v. Hence, we may assume, without loss of generality, that v has indegree at least 2. Thus v has outdegree 1 and the only arc with its tail at v is vu. Since D is arc-minimally strongly connected, there exists $\emptyset \subset S \subset V$ such that $\delta(S) = \{uv\}$. Thus uv is the only arc of D with its tail in S and its head not in S. Furthermore, vu is the only arc with its tail at v so $\delta(S \cup \{v\}) = \emptyset$ and so $S \cup \{v\} = V$. But then $\delta(S)$ is the set of arcs with their heads at v and there are at least 2. Therefore, D cannot have a pair of antiparallel arcs.

Let v be any node of D. Let $C_1, \ldots, C_k$ be the node sets of the components obtained by removing v from the the underlying support graph of D. If v has indegree 1 then there is exactly one edge total in $\delta(C_1) \cup \ldots \cup \delta(C_k)$. If $k \geq 2$ then D is not strongly connected which is a contradiction. Therefore, the underlying support graph of D is 2-vertex-connected. □

In the next section, we look at further properties of the digraphs in $\mathcal{S}$ which relate to a certain constructive characterization of strongly connected digraphs.

8.2 Directed Ear Decompositions

A *directed ear decomposition* is a sequence of directed graphs, $H_0, \ldots H_k$ such that H_0 is a dicycle and H_{i+1} is obtained from H_i by taking a dipath R_{i+1} and identifying the endpoints of R_{i+1} with two (not necessarily distinct) nodes of H_i. This mirrors the well-known notion of ear decompositions for undirected graphs which was first introduced by Whitney [80]. A digraph D is said to have a directed ear decomposition if $D = H_k$ for some directed ear decomposition. Proposition 8.2.1 shows us that strongly connected digraphs are exactly those that have ear decompositions. We provide an outline of the proof for completeness.

Proposition 8.2.1. *A digraph is strongly connected if and only if it has a directed ear decomposition.*

Proof. Suppose that a strongly connected digraph, D, does not have a directed ear decomposition. Notice, that for any arc, uv, of D that there is a (v, u)-dipath in D since D is strongly connected. Hence, D has a dicycle and so let D' be a maximal (with respect to the number of arcs) directed subgraph of D which has a directed ear decomposition, $H_0, \ldots H_k$. If D' contains all the nodes of D then any arc of D that is not in D' can be added to H_k to get a new directed graph H_{k+1}, thereby extending our directed ear decomposition and contradicting the maximality of D'.

Thus, there are nodes of D that are not in D'. Since D is strongly connected, and D' is a directed subgraph of D, there must be an arc uv of D such that u is a node of D', but v is not a node of D'. Let P be any (v, u)-dipath in D and let w be the first node of P, travelling from v to u, which is in D'. Then let H_{k+1} be the digraph obtained by taking H_k and adding the arc uv along with the (v, w)-subdipath of P. However, H_{k+1} has a directed ear decomposition and is a directed subgraph of D, which contradicts the maximality of D'. Therefore, D must have a directed ear decomposition.

To prove that every digraph with a directed ear decomposition is strongly connected, we use induction on the number of digraphs in the directed ear decomposition. Clearly, if the directed ear decomposition has only one digraph, then this digraph is a dicycle which is strongly connected. Otherwise, suppose that a digraph D has a directed ear decomposition $H_0, \ldots, H_k$ such that $H_0, \ldots, H_i$ are all strongly connected, for some $0 \leq i < k$, but H_{i+1} is not strongly connected. Let R_{i+1} be the dipath added to H_i to obtain H_{i+1} and let R_{i+1} be a (u, v)-dipath in H_{i+1}. It is then a simple matter to check that there is an (a, b)-dipath in H_{i+1} for every pair of distinct nodes in H_{i+1}. This contradicts the fact that H_{i+1} is not strongly connected and so we see that every digraph with a directed ear decomposition is strongly connected. □

More to the point, we are interested in the directed ear decompositions of the digraphs in $\mathcal{S}$. We discover certain properties of these ear decompositions as described in Theorem 8.2.2 and use these properties in the next section to actually construct the digraphs of $\mathcal{S}$.

Theorem 8.2.2. *If $D \in \mathcal{S}$ and $H_0, H_1, \dots, H_k$ is a directed ear decomposition of D with dipaths $R_1, \dots, R_k$ then*

1. *H_0 is a dicycle of length at least 3,*
2. *if R_{i+1} is a (u,v)-dipath in H_{i+1} for some $0 \leq i \leq k-1$ then*
   (a) *$u \neq v$ and uv is not an arc of H_i,*
   (b) *R_{i+1} has at least 2 arcs, and*
   (c) *u has indegree 1 and v has outdegree 1 in H_i, and*
3. *$H_i \in \mathcal{S}$ for every $0 \leq i \leq k$.*

Proof. Proposition 8.1.4 tells us that D has no pair of antiparallel arcs so H_0 cannot have length 2. Hence H_0 must have length at least 3.

Notice that $H_0, \dots, H_k$ are directed subgraphs of D. Let R_{i+1} be a (u,v)-dipath in H_{i+1}. If uv is an arc of H_i then since H_{i+1} is a directed subgraph of D, there is the arc uv in D along with an arc-disjoint (u,v)-dipath. Hence any cut separating u and v must contain uv and at least one arc from this dipath. This contradicts the fact that D is arc-minimally strongly connected. The same reasoning applies if R_{i+1} is the single arc uv since H_i is strongly connected and hence must contain a (u,v)-dipath. Thus R_{i+1} must have at least two arcs.

Now, since H_i is strongly connected, u has indegree at least 1 and outdegree at least 1. Thus if $u = v$ then u has indegree at least 2 and outdegree at least 2 in H_{i+1}. Again, H_{i+1} is a directed subgraph of D which contradicts the fact that u either has indegree 1 or outdegree 1 in D. Hence $u \neq v$.

Since H_i is strongly connected, u and v each have indegree at least 1 and outdegree at least 1 in H_i. If u has indegree at least 2 in H_i then it also has indegree at least 2 in D. Furthermore, since we are adding a new (u, v)-dipath to H_i, u has outdegree at least 2 in H_{i+1} and hence in D. This contradicts the fact that u either has indegree 1 or outdegree 1 in D and thus u must have indegree 1 in H_i. Similarly we can argue that v must have outdegree 1 in H_i.

Consider H_i for some $0 \leq i \leq k$. H_i is strongly connected since H_i has the directed ear decomposition $H_0, \ldots, H_i$. If H_i is not arc-minimally strongly connected, then there is some arc uv and some arc-disjoint (u, v)-dipath H_i. However, this arc and dipath are present in D, contradicting the fact that D is arc-minimally strongly connected. Hence H_i must be arc-minimally strongly connected. Lastly, H_i is a directed subgraph of D which is strongly connected and every node of D either has indegree 1 or outdegree 1. Hence the same holds true for H_i. Therefore, $H_i \in \mathcal{S}$. □

In the next section, we use the properties of Theorem 8.2.2 to recursively construct the digraphs of $\mathcal{S}$.

8.3 Constructing the Digraphs of $\mathcal{S}$

Theorem 8.2.2 tells us that if $G, H \in \mathcal{S}$ such that G is obtained from H by adding a directed ear then the directed ear must have its tail and head at nodes of H of indegree 1 and outdegree 1, respectively. Now if we add any directed ear to H with its tail and head at nodes of H of indegree 1 and outdegree 1 respectively then the resulting digraph is certainly strongly connected and has the property that every node has indegree 1 or outdegree 1. However, the resulting digraph may not be in $\mathcal{S}$ if it is not arc-minimally strongly connected. In order to construct the digraphs of $\mathcal{S}$, we would like to find a way to determine whether adding a certain directed ear to H results in an arc-minimally strongly connected digraph.

For this purpose, we require Proposition 8.3.1 below which is the directed version of the "uncrossing lemma" due to Cornuéjols, Fonlupt, and Naddef [25] for the minimum cuts of an undirected graph, and can be proved in a similar manner.

Proposition 8.3.1. *Let $D = (V, E)$ be a strongly connected digraph and let $\emptyset \subset S, T \subset V$. If*

- $S \cap T \neq \emptyset$,
- $S \cup T \neq V$,
- $|\delta(S)| = 1$, *and*
- $|\delta(T)| = 1$

then $|\delta(S \cap T)| = 1$ and $|\delta(S \cup T)| = 1$.

This leads to the following Corollary. We prove just the first part since the proof of the second part is argued in a similar manner.

Corollary 8.3.2. *Let $D = (V, E)$ be an arc-minimally strongly connected digraph. For each $uv \in E$ there is a $\emptyset \subset S_{uv}^{min} \subset V$ such that $\delta(S_{uv}^{min}) = \{uv\}$ and for any $\emptyset \subset T \subset V$ such that $\delta(T) = \{uv\}$ we have that $S_{uv}^{min} \subseteq T$. Likewise there is a $\emptyset \subset S_{uv}^{max} \subset V$ such that $\delta(S_{uv}^{max}) = \{uv\}$ and for any $\emptyset \subset T \subset V$ such that $\delta(T) = \{uv\}$ we have that $T \subseteq S_{uv}^{max}$.*

Proof. Let uv be an arc of D. Since D is an arc-minimally strongly connected digraph, there is some S such that $\delta(S) = \{uv\}$. Notice that $u \in S$ and $v \notin S$. Suppose, without loss of generality, that S is minimal with respect to the number of nodes of S. If T has the property that $\delta(T) = \{u, v\}$, then $u \in T$ and $v \notin T$. Hence, $S \cap T \neq \emptyset$ and $S \cup T \neq V$. Thus, by Proposition 8.3.1, $\delta(S \cap T) = \{uv\}$. However, since S is minimal, it must be that $S \cap T = S$ and so $S \subseteq T$. Hence, the result follows. □

This leads us to a theorem that describes how to build the digraphs of $\mathcal{S}$ using ear decompositions. A related construction for the 2-edge-connected spanning subgraph problem in undirected graphs is presented in [13].

Theorem 8.3.3. *Let $H \in \mathcal{S}$ and let u and v be distinct nodes of H with indegree 1 and outdegree 1 respectively. Let S_{xy}^{min} and S_{xy}^{max} be the subsets of the vertices of H as described in Corollary 8.3.2. Let G be the digraph obtained from H by adding a directed ear, of length at least 2, with its tail at u and its head at v. Then, $G \in \mathcal{S}$ if and only if for every arc, xy, of H either $u \notin S_{xy}^{min}$ or $v \in S_{xy}^{max}$.*

Proof. Let R be the directed ear added to H to obtain G. Let $V(H)$ and $V(G)$ be the node sets of H and G, respectively.

Suppose that there is some arc, xy of H such that $u \in S_{xy}^{\min}$ and $v \notin S_{xy}^{\max}$. Suppose, for a contradiction, that $H[S_{xy}^{\min}]$ does not contain an (x,u)-dipath. Let Q be the set of nodes of $S_{xy}^{\min}$ which are reachable from x in $H[S_{xy}^{\min}]$. Since $u \notin Q$ we know that $Q \subset S_{xy}^{\min}$. However, $\delta(Q) = \{xy\}$ which contradicts the fact that $S_{xy}^{\min}$ is the minimal node set which induces a cut containing only the edge xy. Hence there must be an (x,u)-dipath in $H[S_{xy}^{\min}]$. Similarly we can show that there must be a (v,y)-dipath in $H[V(H)\backslash S_{xy}^{\max}]$. Now $S_{xy}^{\min}$ and $V(H)\backslash S_{xy}^{\max}$ are node disjoint so there is an (x,y)-dipath in G which does not contain the arc xy. Thus $G - xy$ is strongly-connected so G is not arc-minimally strongly-connected.

Hence if $G \in \mathcal{S}$ then G is arc-minimally strongly connected and therefore for every arc xy, of H either $u \notin S_{xy}^{\min}$ or $v \in S_{xy}^{\max}$.

Now suppose that for every arc, xy, of H either $u \notin S_{xy}^{\min}$ or $v \in S_{xy}^{\max}$. Suppose, for a contradiction, that $G \notin \mathcal{S}$. Then G is not arc-minimally strongly-connected and so there must be an arc, xy, such that $G - xy$ is strongly-connected. If xy is an arc of R, then since R contains at least two arcs, either x has indegree 1 and outdegree 1 in G or y has indegree 1 and outdegree 1 in G. Hence $G - xy$ cannot be strongly connected and so xy must be an arc of H.

Since $G - xy$ is strongly-connected, there must be an (x,y)-dipath, P, in G which does not contain xy. Conversely, since $H \in \mathcal{S}$, it must be that $H - xy$ is not strongly-connected and so P must contain some arcs from R. Hence P must contain all the arcs of R and so P contains a (x,u)-dipath which is also in H and a (v,y)-dipath which is also in H. Since H contains an (x, u)-dipath which does not use the arc xy then for any $Q \subset V(H)$ where $\delta(Q) = \{xy\}$ we have $u \in Q$. Thus $u \in S_{xy}^{\min}$. Similarly, for any $Q \subset V(H)$ where $\delta(Q) = \{xy\}$ we have $v \notin Q$. Hence $v \notin S_{xy}^{\max}$. This contradicts our initial assumption and as a result, G must be arc-minimally strongly-connected. Therefore, $G \in \mathcal{S}$. □

In the next section, we see how the SCSSP relates to the ATSP. To quantify this relationship, we use directed ear decompositions of the digraphs in $\mathcal{S}$.

8.4 Relating the SCSSP to the ATSP

In this section, we compare the SCSSP to the ATSP and also other related problems. In addition, we consider the relationships between the linear programming relaxations of these various problems. Specifically, we discover that the relationship between the SCSSP and the ATSP differs greatly from the relationship between their respective undirected counterparts. In fact, we see that the worst-case ratio between the ATSP and the SCSSP can be as bad as $n - 2$, even for metric cost functions. This leads us to some consequences about how far we can relax the ATSP and hope for a constant bound on the optimal value of the ATSP.

First, we formulate the SCSSP as the integer program

$$
\begin{array}{lrcll}
\text{Minimize} & cx & & & \\
 & x(\delta(S)) & \geq & 1 & \text{for all } \emptyset \subset S \subset V \\
 & x_e & \geq & 0 & \text{for all } e \in E \\
 & x & \in & \mathbb{Z}^E. &
\end{array}
$$

Notice that this integer programming formulation of the SCSSP can be obtained from that of the ATSP by simply relaxing the node equalities.

Just as with the ATSP, we take a linear programming relaxation of this integer program by simply removing the integrality condition.

$$\begin{array}{lrcll} \text{Minimize} & cx & & & \\ & x(\delta(S)) & \geq & 1 & \text{for all } \emptyset \subset S \subset V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

We use $\text{SCSSP}_{LP}(c)$ to denote the optimal value of this linear program (and also when the context is clear, the linear program itself).

Despite the similarity between the ATSP and the SCSSP, they differ greatly in our knowledge of their respective integrality gaps. As previously discussed in this paper, there is no known upper bound on the integrality gap of the ATSP to its linear programming relaxation, even when we require the cost functions to be metric. Conversely, Frederickson and Ja'Ja' [34] show that

$$\frac{\text{SCSSP}(c)}{\text{SCSSP}_{LP}(c)} \leq 2$$

for any (not necessarily metric) nonnegative cost function c. In fact, they provide a 2-approximation algorithm for the SCSSP. Their approximation algorithm involves finding a minimum-cost arborescence rooted at a node r which is an optimal solution to the linear program

$$\begin{array}{lrcll} \text{Minimize} & cx & & & \\ \text{Subject to} & x(\delta(S)) & \geq & 1 & \text{for all } \{r\} \subseteq S \subset V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

Edmonds [29] showed that the feasible region of this linear program is in fact an integral polyhedron. Frederickson and Ja'Ja' then found a minimum-cost reverse arborescence rooted at r which is an optimal solution to the linear program

$$\begin{array}{lrcll} \text{Minimize} & cx & & & \\ \text{Subject to} & x(\delta(S)) & \geq & 1 & \text{for all } \emptyset \subset S \subseteq V\backslash\{r\} \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

Notice that both these linear programs are relaxations of the $\text{SCSSP}_{LP}(c)$. If we let $T^+(c)$ and $T^-(c)$ denote the optimal values of these linear programs, then we have

$$T^+(c) \leq \text{SCSSP}_{LP}(c)$$

and

$$T^-(c) \leq \text{SCSSP}_{LP}(c).$$

However, the union of a minimum-cost arborescence rooted at r and a minimum-cost reverse arborescence rooted at r is a strongly connected spanning subgraphs. Thus,

$$T^+(c) + T^-(c) \geq \text{SCSSP}(c)$$

and we have

$$\frac{\text{SCSSP}(c)}{\text{SCSSP}_{LP}(c)} \leq 2.$$

More recently, Melkonian and Tardos [64] provide a primal-dual-based 3-approximation algorithm for the $\text{SCSSP}(c)$ for arbitrary nonnegative costs, c. They conjecture, however, that their algorithm is in fact a $(2 - 2/n)$-approximation algorithm. Since they use $\text{SCSSP}_{LP}(c)$ to prove the worst-case ratio, this implies the conjecture

$$\frac{\text{SCSSP}(c)}{\text{SCSSP}_{LP}(c)} \leq 2 - \frac{2}{n}.$$

For the undirected analogue of this problem, Alexander, Boyd, and Elliott-Magwood [1] remark that the integrality gap of the 2-edge-connected spanning subgraph problem to its linear programming relaxation is at most $3/2$.

We now turn our attention to a result about assigning costs to the digraphs of $\mathcal{S}$. We see that the digraphs in $\mathcal{S}$ are also optimal solutions for certain instances of the linear programming relaxation of the SCSSP.

Theorem 8.4.1. *Let $D \in \mathcal{S}$ and let c' be any set of arc-costs assigned to the arcs of D. Let c be the metric completion of c'. Then the characteristic vector of D is an optimal solution to $SCSSP_{LP}(c)$.*

Proof. Notice that since D is arc-minimally strongly-connected, for any arc, uv, of D, the arc uv is the only (u, v)-dipath in D. Hence c'_{uv} is the cost of a min-cost (u, v)-dipath in D. Thus c is metric.

Let x be any optimal solution to the LP relaxation of the SCSSP with respect to c. If $x_{uv} > 0$ for some arc which is not an arc of D, we can find a min-cost (u, v)-dipath in D and reroute the flow on the arc uv through this dipath. The resulting set of arc-values is a feasible solution to the linear relaxation of cost cx and hence is an optimal solution. Thus we may assume that the support digraph of x is a directed subgraph of D.

Let $V(D)$ denote the node set of D and let $E(D)$ be its arc set.

Now D is arc-minimally strongly-connected so for every $uv \in E(D)$ there exists a $Q \subset V(D)$ such that $\delta(Q) = \{uv\}$. Since x is a solution to the LP relaxation, we have $x(\delta(Q)) \geq 1$. Thus $x_{uv} \geq 1$. Hence for every $uv \in E(D)$ we have $x_{uv} \geq 1$. Thus

$$\sum_{e \in E(D)} c_e \leq \sum_{e \in E(D)} c_e x_e$$

and so D is an optimal solution to the linear programming relaxation of the SCSSP.

□

The next result shows that the ratio of $\text{ATSP}_{LP}(c)$ to $\text{SCSSP}(c)$ can be at least $n - 2$ for a metric cost function c. This contrasts sharply with the analogous result for the undirected problems. Williamson [81] and Bertsimas and Goemans [8] independently show that when c is metric, the optimal value of the linear programming relaxation of the STSP is equal to the optimal value of the linear programming relaxation of the 2-edge-connected spanning subgraph problem. Hence, the optimal value of the linear programming relaxation of the STSP is less than, or equal to, the optimal value of the 2-edge-connected spanning subgraph problem.

Proposition 8.4.2. *For every $n \geq 3$, there is a metric set of arc-costs, c, on the complete digraph on n nodes such that*

$$\frac{ATSP_{LP}(c)}{SCSSP(c)} = n - 2$$

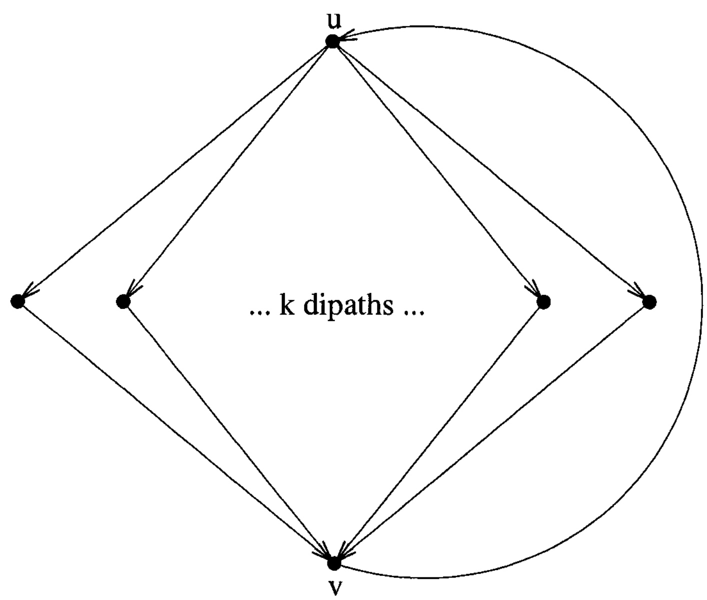

Figure 8.3: The digraph D_k

and

$$\frac{ATSP(c)}{SCSSP(c)} = n - 2.$$

Proof. Consider the digraph shown in Figure 8.3 with $k = n - 2$ (u, v)-dipaths each of length 2. Call this digraph D_k.

If we assign a cost of 1 to the arc vu and 0 to all the other arcs of D_k then the cost of D_k is 1. The metric completion, c, of this set of costs is obtained by assigning a cost of 0 to the arc uv and 1 to all remaining arcs which are not in D. By Theorem 8.4.1, D is an optimal solution to the SCSSP(c).

Let x be any optimal solution of ATSP$_{LP}(c)$. We reroute x through D (as we did in the proof of Theorem 8.4.1) and so we may assume that x satisfies $x(\delta(v)) = x(\delta(V\backslash\{v\}))$ for every $v \in V$ and that $x(\delta(S)) \geq 1$ for every proper non-empty subset S of the node set of D. Now each node, w, of D apart from u or v must have $x(\delta(w)) \geq 1$ and so $x(\delta(V\backslash\{v\})) \geq k$. Hence, $x(\delta(v)) \geq k$ and so $x_{uv} \geq k$. Thus ATSP$_{LP}(c) = cx \geq k$.

Conversely, if we assign a flow of k to uv and 1 to all remaining arcs of D, we get the characteristic vector of a directed Eulerian multigraph of cost k. We can then shortcut this Eulerian multi-digraph to get a tour of cost k and so we know that $\text{ATSP}_{LP}(c) \leq k$. Thus

$$\frac{\text{ATSP}_{LP}(c)}{\text{SCSSP}(c)} = k.$$

Notice however, that the tour described above corresponds to an integer optimal solution. Hence

$$\frac{\text{ATSP}(c)}{\text{SCSSP}(c)} = k.$$

□

Thus we have shown that the ratio of the ATSP to the SCSSP can be at least $n-2$ when c is metric. We later show that this is the worst case. For now, we explore another related problem.

Given a set of nonnegative arc costs on the arcs of the complete digraph on n nodes, consider the problem of finding a minimum-cost strongly connected spanning sub-multigraph which also has the property that each node in the solution has the same indegree as its outdegree. We call this the *Asymmetric Graphical Travelling Salesman Problem* and we denote the optimal value (and sometimes the problem itself when the context warrants) of an instance of this problem by AGTSP(c). We can formulate this problem as an integer program, namely,

$$\begin{array}{llll}
\text{Minimize} & cx & & \\
& x(\delta(v)) & = & x(\delta(V\backslash\{v\})) \quad \text{for all } v \in V \\
& x(\delta(S)) & \geq & 1 \quad \text{for all } \emptyset \subset S \subset V \\
& x_e & \geq & 0 \quad \text{for all } e \in E \\
& x_e & \in & \mathbb{Z}^E.
\end{array}$$

Notice that a feasible solution to this integer program is the characteristic vector of a graphical tour (as presented in chapter 5). If the costs are metric, then by shortcutting as described in Lemma 5.2.1 we can take an optimal solution of the AGTSP and get an optimal solution of the ATSP of the same cost. Since the AGTSP is a relaxation of the ATSP we have AGTSP(c) = ATSP(c) for all metric cost functions c.

Consider a linear programming relaxation of the AGTSP, namely

$$\begin{array}{lrcll} \text{Minimize} & cx & & & \\ \text{Subject to} & x(\delta(v)) & = & x(\delta(V\backslash\{v\})) & \text{for all } v \in V \\ & x(\delta(S)) & \geq & 1 & \text{for all } \emptyset \subset S \subset V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

We use AGTSP$_{LP}(c)$ to denote the optimal value (and at times the problem itself) of this linear program. Notice that AGTSP$_{LP}(c)$ is a relaxation of ATSP$_{LP}(c)$ and hence AGTSP$_{LP}(c) \leq$ ATSP$_{LP}(c)$ for all c. In fact, Goemans [38] shows in Theorem 6.1 of his thesis that AGTSP$_{LP}(c)$ = ATSP$_{LP}(c)$ whenever c is metric.

We now prove that the ratio of AGTSP$_{LP}(c)$ to SCSSP(c) is at most $n-2$ for any metric cost function c. This then implies the same result for the ratio of ATSP$_{LP}(c)$ to SCSSP(c).

Proposition 8.4.3. *For every $n \geq 3$ and every set of metric arc-costs, c, on the complete digraph on n nodes*

$$\frac{AGTSP_{LP}(c)}{SCSSP(c)} \leq n-2$$

and

$$\frac{AGTSP(c)}{SCSSP(c)} \leq n-2.$$

Proof. Let $D \in \mathcal{S}$ be the support digraph of an optimal solution of SCSSP(c) on the complete digraph (V, E). Let $E(D)$ be the arc-set of D. Let c' be the metric completion of c restricted to $E(D)$. Since c is metric, AGTSP$_{LP}(c) \leq$ AGTSP$_{LP}(c')$ whereas SCSSP(c) = SCSSP(c'). Thus

$$\frac{\text{AGTSP}_{LP}(c)}{\text{SCSSP}(c)} \leq \frac{\text{AGTSP}_{LP}(c')}{\text{SCSSP}(c')},$$

and so we may assume, without loss of generality, that $c = c'$. Thus, we may reroute our optimal solution, x, of $\text{AGTSP}_{LP}(c)$ through D to get another optimal solution to $\text{AGTSP}_{LP}(c)$. Thus, we may assume that the support digraph of x is a directed subgraph of D.

Furthermore, since D is arc-minimally strongly connected, every $e \in E(D)$ is the unique arc in some tight cut. However, $x(\delta(S)) \geq 1$ for every $\emptyset \subset S \subset V$ and hence $x_e \geq 1$ for every $e \in E(D)$. Thus $\text{SCSSP}(c) \leq \text{AGTSP}_{LP}(c)$.

We now proceed by induction on the number of directed subgraphs in a directed ear decomposition of D. If D is a directed cycle, then the characteristic vector of D is also a feasible solution to $\text{AGTSP}_{LP}(c)$. However, $\text{SCSSP}(c) \leq \text{AGTSP}_{LP}(c)$ and so $\text{SCSSP}(c) = \text{AGTSP}_{LP}(c)$. Thus the result holds if there is only one directed subgraph in a directed ear decomposition of D.

Hence, let D be a digraph with r directed subgraphs in its directed ear decomposition where $r \geq 2$. Now, suppose that the result holds for all digraphs with fewer than r directed subgraphs in a directed ear decomposition.

Let Q be the last ear added in the ear decomposition and let H be the directed subgraph of D that Q was added to to obtain H. Since D is arc-minimally strongly connected, we know that $|Q| \geq 2$, and by Theorem 8.2.2 we know that $H \in \mathcal{S}$.

Let c_H be the set of costs on the arcs of the complete digraph on the nodes of H defined as follows. Restrict c to the arcs of H and take the metric completion of the resulting arc-costs. Thus $\text{SCSSP}(c) = \text{SCSSP}(c_H) + c(Q)$. Now suppose Q is a (u,v)-dipath in D. Since H is strongly connected, let P be a (v,u)-dipath of H of minimum cost with respect to c_H. Now let y be any optimal solution of $\text{AGTSP}_{LP}(c_H)$. Define a feasible solution, x', to $\text{AGTSP}_{LP}(c)$ by

$$x'_e = \begin{cases} y_e & \text{if } e \in E(H) \backslash P \\ y_e + 1 & \text{if } e \in P \\ 1 & \text{if } e \in Q \\ 0 & \text{otherwise.} \end{cases}$$

Essentially, we are taking y and adding 1 to the values of all arcs on the dicycle composed of the union of P and Q. As a result, x' is a feasible solution to $\text{AGTSP}_{LP}(c)$. Thus,

$$\begin{aligned} \text{AGTSP}_{LP}(c) &\leq cx' \\ &= c_H y + c(P) + c(Q) \\ &= \text{AGTSP}_{LP}(c_H) + c(P) + c(Q). \end{aligned}$$

Now, H has $n - |Q| + 1$ nodes where $|Q| \geq 2$ and H has a directed ear decomposition containing $r - 1$ directed subgraphs, so by our induction hypothesis, $\text{AGTSP}_{LP}(c_H) \leq (n - |Q| + 1 - 2)\text{SCSSP}(c_H)$. Putting all the pieces together, we get

$$\begin{aligned} &\frac{\text{AGTSP}_{LP}(c)}{\text{SCSSP}(c)} \\ &\leq \frac{(n - |Q| - 1)\text{SCSSP}(c_H) + c(P) + c(Q)}{\text{SCSSP}(c_H) + c(Q)} \\ &= \frac{(n - |Q| - 1)\text{SCSSP}(c_H) + (n - |Q| - 1)c(Q) + c(P) - (n - |Q| - 2)c(Q)}{\text{SCSSP}(c_H) + c(Q)} \\ &= (n - |Q| - 1) + \frac{c(P) - (n - |Q| - 2)c(Q)}{\text{SCSSP}(c_H) + c(Q)} \\ &\leq (n - |Q| - 1) + \frac{c(P)}{\text{SCSSP}(c_H)} \qquad \text{since } c(Q) \geq 0 \\ &\leq n - |Q| - 1 + 1 \qquad \text{since } c(P) \leq \text{SCSSP}(c_H) \\ &= n - |Q| \\ &\leq n - 2 \qquad \text{since } |Q| \geq 2 \end{aligned}$$

Hence the result follows by induction.

Notice that none of our arguments change if we assume that y is integer and by our induction hypothesis we have that $\text{AGTSP}(c_H) \leq (n - |Q| + 1 - 2)\text{SCSSP}(c_H)$. Thus, by the same reasoning,

$$\frac{\text{AGTSP}(c)}{\text{SCSSP}(c)} \leq n - 2.$$

□

We have stated before that if c is metric then AGTSP(c) = ATSP(c). Furthermore, Goemans [38] shows that if c is metric then AGTSP$_{LP}$(c) = ATSP$_{LP}$(c). This leads immediately, using Proposition 8.4.3, to the following corollary.

Corollary 8.4.4. *For every $n \geq 3$ and every set of metric arc-costs, c, on the complete digraph on n nodes*

$$\frac{ATSP_{LP}(c)}{SCSSP(c)} \leq n - 2$$

and

$$\frac{ATSP(c)}{SCSSP(c)} \leq n - 2.$$

□

In addition, by Proposition 8.4.2, we see that these upper bounds are tight. By a careful analysis of the proof of Proposition 8.4.3 we see that the family (call it $\mathcal{F}$) of digraphs described in Proposition 8.4.2, and depicted in Figure 8.3, are the unique digraphs of $\mathcal{S}$ which attain the upper bound.

The result in Corollary 8.4.4 is startling since Cunningham (see [66]) and Bertsimas and Goemans [8] show that the undirected analogue of the first part of the corollary has an upper bound of 1. Furthermore, Monma, Munson, and Pulleyblank [66] show that the undirected analogue of the second part of the corollary has an upper bound of $\frac{4}{3}$.

Now consider the linear programming formulation of AGTSP$_{LP}$(c) given by

$$\begin{array}{lrcll} \text{Minimize} & cx & & & \\ \text{Subject to} & x(\delta(v)) & = & x(\delta(V \backslash \{v\})) & \text{for all } v \in V \\ & x(\delta(S)) & \geq & 1 & \text{for all } \emptyset \subset S \subset V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

For any $w \in V$, the node equality $x(\delta(w)) = x(\delta(V \backslash \{w\}))$ is implied by all the remaining node equalities. This can be demonstrated by observing that for assignment of values, x, to the arcs in E,

$$\begin{aligned} \sum_{v \in V} x(\delta(v)) &= \sum_{v \in V} x(\delta(V \backslash \{v\})) \\ x(\delta(w)) + \sum_{v \in V \backslash \{w\}} x(\delta(v)) &= x(\delta(V \backslash \{w\})) + \sum_{v \in V \backslash \{w\}} x(\delta(V \backslash \{v\})). \end{aligned}$$

Thus, we can remove one of the node equalities from the definition of $\text{AGTSP}_{LP}(c)$ without changing the feasible solutions of this linear program. However, if we remove any two node equalities, then the digraphs in the family $\mathcal{F}$ are feasible solutions to the linear program and hence the ratio between $\text{AGTSP}_{LP}(c)$ and the optimal value of any such relaxation is $n-2$. On the other hand, if for some $\emptyset \subset S \subset V$ we remove the restriction that $x(\delta(S)) \geq 1$ and $x(\delta(V \backslash S)) \geq 1$ then we can find some feasible solution, y, to this relaxation where $y(\delta(S)) = 0$. Let c be any metric set of arc-costs and let x be any feasible solution of $\text{AGTSP}_{LP}(c)$. Let c' be the set of arc-costs given by

$$c'_e = \begin{cases} c_e + \epsilon & \text{if } e \in \delta(S) \\ c_e & \text{otherwise} \end{cases}$$

where ϵ is some nonnegative real number. It is straightforward to check that c' is metric and

$$\begin{aligned} \frac{c'x}{c'y} &= \frac{cx + \epsilon x(\delta(S))}{cy + \epsilon y(\delta(S))} \\ &\geq \frac{cx + \epsilon}{cy}. \end{aligned}$$

Thus, by increasing the value of ϵ, we can create sets of arc-costs which make the ratio of $\text{AGTSP}_{LP}(c')$ to the optimal value of the relaxation as large as we would like. Notice that this ratio is ∞ and does not depend on n. This result holds for a more general class of linear programs. Any linear program which contains the cut constraints ($x(\delta(S)) \geq 1$ for every $\emptyset \subset S \subset V$) can have an arbitrarily large optimal value when compared to a linear program which has a feasible solution, y, where $y(\delta(S)) = 0$ for some $\emptyset \subset S \subset V$.

Hence, if we remove constraints from the linear programming formulation of $\text{AGTSP}_{LP}(c)$ to obtain a certain relaxation then the ratio of $\text{AGTSP}_{LP}(c)$ to the optimal value of the relaxation is either $n-2$ or ∞.

Some famous linear programs which would have this ratio of infinity are the directed arborescence problem and the directed 2-factor problem (whose feasible region is the assignment polytope). Given a node $r \in V$, the directed arborescence problem is

$$\begin{array}{lrcll} \text{Minimize} & cx & & & \\ \text{Subject to} & x(\delta(S)) & \geq & 1 & \text{for all } \{r\} \subseteq S \subset V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

The directed 2-factor problem is

$$\begin{array}{lrcll} \text{Minimize} & cx & & & \\ \text{Subject to} & x(\delta(v)) & = & x(\delta(V \backslash \{v\})) & \text{for all } v \in V \\ & x_e & \geq & 0 & \text{for all } e \in E \\ & x & \in & \mathbb{R}^E. & \end{array}$$

Thus, we really cannot remove any constraints from AGTSP_{LP} (except one of the node constraints) and expect to get a guarantee of a reasonable lower bound on the optimal value of the ATSP (or even the ATSP_{LP}). In this sense, the linear programming relaxation of the Asymmetric Graphical Travelling Salesman Problem has a minimal set of constraints that might yield a reasonable lower bound on the ATSP with metric costs. Of course, we still do not know if $\text{AGTSP}_{LP}(c)$ itself is a good lower bound on $\text{ATSP}(c)$ for metric c since we do not know of any upper bound on the integrality gap for the ATSP.

n	ASEP	AGTSP$_{LP}$	SCSSP$_{LP}$
3	2	5	5
4	12	64	64
5	384	2209	2389
6	57720	251496	389976

Table 8.1: Number of extreme points

n	AGTSP$_{LP}$	SCSSP$_{LP}$
3	5	6
4	20	12
5	84	20
6	409	30

Table 8.2: Number of extreme rays

8.5 Extreme Points of Polyhedra related to the ASEP

We finish off this chapter with a brief look at the number of extreme points and extreme rays of two polyhedra which are related to the ASEP. Specifically, we look at the polyhedron given by the feasible region of the AGTSP$_{LP}$ and the polyhedron given by the feasible region of the SCSSP$_{LP}$. Neither of these polyheda are polytopes because the characteristic vector of a dicycle in K_n is a ray of each polyhedron.

To find the extreme points and extreme rays of each polyhedron, we use CDD+ [36]. We do not remove isomorphic copies of extreme points or extreme rays. The results are summarized in Table 8.1 and Table 8.2.

Chapter 9

Conclusions

9.1 Results

In this section, we summarize the results of each chapter of the thesis. We highlight our main accomplishments and new results.

In Chapter 3, we report on some well-known results about the structure of ASEP extreme points. We then present the operation of inserting a 1-arc for generating new extreme points of the ASEP. We also present the 2-jack operation for generating new half-integer extreme points of the ASEP. We see that, using these operations, we can generate many of the extreme points of the ASEP for small values of n.

In Chapter 4, we formulate the computation of the integrality gap for a given extreme point as a linear program. By finding all the extreme points of the ASEP for $4 \leq n \leq 7$ we are able to compute the integrality gap exactly for each of these values of n. We then find a subset of the extreme points of the ASEP for $8 \leq n \leq 9$ and use these to compute a lower bound on the integrality gap for these values of n. We also prove the result that the 1-arc insertion operation never increases the integrality gap. This allows us to ignore any extreme point with a 1-arc when we are using the framework presented in the thesis to find extreme points with large integrality gaps.

In Chapter 5, we present two families of half-integer extreme points. These two families are based on the patterns we see in the extreme points which attain large integrality gaps as seen in Chapter 4. The first family attains an integrality gap arbitrarily close to 3/2 whereas the second family attains an integrality gap arbitrarily close to 2 (the largest possible integrality gap for a half-integer extreme point of the ASEP). The first family refutes the conjecture that the integrality gap of the ASEP is 4/3 [16]. We compare the second family to that presented by Charikar, Goemans, and Karloff [18] and show that our family had certain advantages.

In Chapter 6, we define the strength of a facet of the ATSP-polytope. We then proceed to calculate the strength of several well-known facets. This measure of strength is also an integrality gap of a certain related metric cost function. We do not find any strength greater than 6/5.

In Chapter 7, we notice that a large number of the extreme points of the ASEP have a "unique" cobasis. We exploit this fact in a pivoting algorithm which enumerates a large number of the non-isomorphic extreme points of the ASEP for $4 \leq n \leq 7$. In all cases, we are able to generate all but at most three extreme points of the ASEP using our algorithm.

In Chapter 8, we study the SCSSP and the AGTSP. We give a structural characterization of a minimal set of optimal solutions to the SCSSP with respect to metric cost functions. We show that the ratio of the AGTSP to the SCSSP is proportional to n. Furthermore, we show that the bound we discover is tight. We also note that relaxing more than one of the equalities of the AGTSP would have as large a ratio with the AGTSP.

9.2 Further Work

Each chapter of the thesis inspires new questions that can be investigated by interested researchers. We discuss some of the many questions that are left unanswered.

In Chapter 3, we present two operations for generating new extreme points of the ASEP. Both of these operations are simple and based on local structural modifications despite the fact that extreme points are defined by systems of linear equations. Are there other operations which can be used to generate new extreme points, especially for those that are not half-integer? It would also be interesting to investigate whether extreme points of the ASEP on $n+1$ nodes obtained by performing the 1-arc insertion or 2-jack operation to the same extreme point of the ASEP on n nodes are adjacent. If so, this might lead to a combinatorial pivoting algorithm for visiting the extreme points of the ASEP.

In Chapter 4, we present the effect of the 1-arc insertion operation on the integrality gap. It would be interesting to see if we could also pinpoint the exact effect of the 2-jack operation on the integrality gap. Of course, the clear open question suggested in this chapter is that of finding a much faster way of finding the non-isomorphic extreme points of the ASEP. The methods presented in Chapter 4 do not take into account the symmetry of the solutions and hence we end up generating many redundant extreme points. We provide a possible solution to this problem in Chapter 7 and we discuss these ideas further there.

In Chapter 5, we present two families of half-integer extreme points which attain large integrality gaps. Certainly, if we want to show that the integrality gap is larger than 2, then we have to find a family of non-half-integer extreme points to prove it. Is there a pattern among the non-half-integer extreme points that we found in Chapter 4 that helps us in this endeavour? Conversely, if we want to show that the integrality gap is exactly 2, then is there a way of showing that the second family of extreme points presented in Chapter 5 produce the largest possible integrality gaps for their given value of n? As we see at the beginning of this chapter, the overwhelming majority of extreme points for the ASEP seem to be non-half-integer and so a further exploration of these interesting extreme points is necessary to reach any reasonable conclusion about the integrality gap.

In Chapter 6, we discuss the strength of facets. It is our hope that the strengths we find would be greater and hence give us a new direction in our search for large integrality gaps. We attempted to prove that the integrality gap for a given value of n must be equal to the strength of some facet of the ATSP-polytope on n nodes but were unsuccessful. As well, the large integrality gaps attained by the second family of extreme points found in Chapter 5 show us that the associated cost functions are not equivalent to any facet-inducing inequality studied in Chapter 6. Do these cost functions then define previously unknown facets of the ATSP-polytope (could we check for small values of n)? In the same line of thinking, can we use the operations presented in Chapter 4 to discover previously unknown facets of the ATSP-polytope? It would be interesting to see what effect lifting operations for facets have on the strength of the facet. It would also be interesting to see if the concept of strength can be used to draw some conclusions about the effectiveness of using certain facets in cutting plane algorithms for solving the ATSP.

In Chapter 7, we introduce a new algorithm for generating ASEP extreme points which took into account isomorphic symmetry and also the "uniqueness" of the cobases of many ASEP extreme points. We would like to extend our results to $n = 8$ and beyond. It would be interesting to see if we could define a canonical cobasis for each extreme point. Hopefully, the structure of this canonical cobasis could greatly speed up the generation of extreme points. Also, it would be helpful to see if we could incorporate the ideas of Avis and Fukada [3] into our algorithm. Lastly, if we could discover an invertible recursive definition for the extreme points of the ASEP on n nodes using only information about the extreme points of the ASEP on fewer than n nodes then we could use the ideas of McKay [62] to generate only one candidate from each isomorphic equivalence class of the extreme points.

In Chapter 8, we state the previously known result that the integrality gap of the SCSSP is at most 2. Melkonian and Tardos [64] conjecture that this integrality gap is at most $2 - 2/n$. We generate all the extreme points of the SCSSP_{LP} for $3 \leq n \leq 6$. These could be used, along with a similar method as computing the integrality gap of the ATSP, to compute the integrality gap of the SCSSP for small values of n.

Bibliography

[1] Anthony Alexander, Sylvia Boyd, and Paul Elliott-Magwood, "On the Integrality Gap of the 2-Edge Connected Subgraph Problem", *Technical Report* (TR-2006-04), University of Ottawa, School of Information Technology and Engineering, 2006.

[2] Norbert Ascheuer, Matteo Fischetti, and Martin Grötschel, "Solving the Asymmetric Travelling Salesman Problem with time windows by branch-and-cut", *Mathematical Programming*, **90** (2001), 475 - 506.

[3] David Avis and Komei Fukuda, "A Pivoting Algorithm for Convex Hulls and Vertex Enumeration of Arrangements and Polyhedra", *Discrete Computational Geometry*, **8** (1992), 295 - 313.

[4] Egon Balas and Matteo Fischetti, "A lifting procedure for the asymmetric traveling salesman polytope and a large new class of facets", *Mathematical Programming*, **58** (1993), 325 - 352.

[5] M. L. Balinski and Andrew Russakoff, "On the assignment polytope", *SIAM Review*, **16** (1974), 516 - 525.

[6] H. G. Bartels and S. G. Bartels, "The Facets of the Asymmetric 5-City Traveling Salesman Polytope", *Methods and Models of Operations Research*, **33** (1989), 193 - 197.

[7] M. Bellmore and J. C. Malone, "Pathology of traveling-salesman subtour-elimination algorithms", *Operations Research*, **19** (1971), 278 - 307.

[8] Dimitris J. Bertsimas and Michel X. Goemans, "On the Parsimonious Property of Connectivity Problems", *Proceedings of the first annual ACM-SIAM symposium on Discrete algorithms*, (1990), 388 - 396.

[9] G. D. Birkhoff, "Tres observaciones sobre el algebra lineal", *Univ. Nac. Tucumán Rev. Ser. A*, **5** (1946), 147 - 151.

[10] Markus Bläser, "A New Approximation Algorithm for the Asymmetric TSP with Triangle Inequality", *Proceedings of the* 14^{th} *annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2003, 638 - 648.

[11] J. A. Bondy and U. S. R. Murty, *Graph theory with applications*, Elsevier Science Publishing Co., Inc., New York, 1976.

[12] K. -H. Borgwardt, "The Average Number of Pivot Steps Required by the Simplex-Method is Polynomial", *Zeitschrift für Operations Research*, **26** (1982), 157 - 177.

[13] Sylvia Boyd and Paul Elliott-Magwood, "Constructing Minimum Cost 2-Edge Connected Spanning Subgraphs of Metric Cost Functions", *Technical Report* (TR-2006-03), University of Ottawa, School of Information Technology and Engineering, 2006.

[14] Sylvia Boyd and Geneviève Labonté, "Finding the exact integrality gap for small Travelling Salesman Problems", *Integer Programming and Combinatorial Optimization:* 9^{th} *International IPCO Conference* in Lecture Notes in Computer Science, **2337** (2002), 83 - 92.

[15] S. C. Boyd and W. R. Pulleyblank, "Optimizing over the subtour polytope of the travelling salesman problem", *Mathematical Programming*, **49** (1991), 163 - 187.

[16] Robert Carr and Santosh Vempala, "On the Held-Karp relaxation for the asymmetric and symmetric traveling salesman problems", *Math Programming, Series A*, **100** (2004), 569 - 587.

[17] L. Sunil Chandran and L. Shankar Ram, "Approximations for ATSP with parametrized triangle inequality", *Proceedings of the* 19^{th} *International Symposium on Theoretical Aspects of Computer Science (STACS)* in Lecture Notes in Computer Science, **2285** (2005), 227 - 237.

[18] Moses Charikar, Michel X. Goemans, Howard Karloff, "On the Integrality Ratio for Asymmetric TSP", *Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, 2004, 101 - 107.

[19] Kevin K. H. Cheung, William H. Cunningham, and Lawrence Tang, "Optimal 3-terminal cuts and linear programming", *Mathematical Programming, Series A*, **106** (2006), 1 - 23.

[20] Sunil Chopra, "Polyhedra of the equivalent subgraph problem and some edge connectivity problems", *SIAM Journal of Discrete Mathematics*, **5** (1992), 321 - 337.

[21] Thomas Christof and Andreas Löbel, *POlyhedron Representation Transformation Algorithm (PORTA)*, version 1.4.0, http://www.zib.de/Optimization/Software/Porta/, copyright 1997.

[22] N. Christofides, "Worst-case Analysis of a New Heuristic for the Traveling Salesman Problem", Technical Report, GSIA, Carnegie Mellon University, 1976.

[23] V. Chvátal, "Edmonds polytopes and a hierarchy of combinatorial problems", *Discrete Mathematics*, **4** (1973), 305 - 337.

[24] V. Chvátal, "Edmonds polytopes and weakly hamiltonian graphs", *Mathematical Programming*, **5** (1973), 29 - 40.

[25] Gérard Cornuéjols, Jean Fonlupt, and Denis Naddef, "The traveling salesman problem on a graph and some related integer polyhedra", *Mathematical Programming*, **33** (1985), 1 - 27.

[26] George B. Dantzig, "Maximization of a linear function of variables subject to linear inequalities", in *Activity Analysis of Production and Allocation*, ed. T. C. Koopmans, Wiley, New York, 1951, 339 - 347.

[27] G. Dantzig, R. Fulkerson, and S. Johnson, "Solution of a large-scale traveling-salesman problem", *Operations Research*, **2** (1954), 393 - 410.

[28] W. L. Eastman, "Linear Programming with Pattern Constraints", Ph. D. thesis, *Harvard University*, Cambridge, 1958.

[29] Jack Edmonds, "Optimal Branchings", *Journal of Research of the National Bureau of Standards*, **71B** (1967), 233 - 240.

[30] R. Euler and H. Le Verge, "Complete linear descriptions of small asymmetric traveling salesman polytopes", *Discrete Applied Mathematics*, **62** (1995), 193 - 208.

[31] M. Fischetti, "Facets of the Asymmetric Traveling Salesman Polytope", *Mathematics of Operations Research*, **16** (1991), 42 - 56.

[32] Matteo Fischetti, "Clique tree inequalities define facets of the asymmetric traveling salesman polytope", *Discrete Applied Mathematics*, **56** (1995), 9 - 18.

[33] Matteo Fischetti and Paolo Toth, "A Polyhedral Approach to the Asymmetric Traveling Salesman Problem", *Management Science*, **43** (1997), 1520 - 1536.

[34] Greg N. Frederickson and Joseph Ja'Ja', "Approximation algorithms for several graph augmentation problems", *SIAM Jounal on Computing*, **10** (1981), 270 - 283.

[35] A. M. Frieze, G. Galbiati, and F. Maffioli, "On the Worst-Case Performance of Some Algorithms for the Asymmetric Traveling Salesman Problem", *Networks*, **12** (1982), 23 - 39.

[36] Komei Fukuda, *cddr+*, version 0.76, http://www.cs.mcgill.ca/~fukuda/soft/cddman/cddman.html, 1999.

[37] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness*, W. H. Freeman, San Francisco, 1979.

[38] Michel X. Goemans, "Analysis of linear programming relaxations for a class of connectivity problems", *Ph. D. Thesis*, Massachusetts Institute of Technology, Sloan School of Management, 1990, http://hdl.handle.net/1721.1/13595.

[39] Michel X. Goemans, "Worst-case Comparison of Valid Inequalities for the TSP", *Mathematical Programming*, **69** (1995), 335 - 349.

[40] Michel X. Goemans, "Minimum Bounded Degree Spanning Trees", *Proceedings of the* 47^{th} *Annual IEEE Symposium on Foundations of Computer Science*, 2006, 273 - 282.

[41] David Goldberg, Christopher Malon, and Marshall Bern, "A global approach to automatic solution of jigsaw puzzles", *Computational Geometry*, **28** (2004), 165 - 174.

[42] R. E. Gomory, "Solving linear programming problems in integers", in: *Combinatorial Analysis* (R. Bellman and M. Hall, eds.), *Proceedings of Symposia in*

Applied Mathematics X, American Mathematical Society, Providence, 1960, 211 - 215.

[43] Martin Grötschel, *Polyedrische Charakterisierungen kombinatorischer Optimierungsprobleme, Mathematical Systems in Economics 36*, Verlag Anton Hain, 1977.

[44] Martin Grötschel and Manfred W. Padberg, "Partial linear characterizations of the asymmetric travelling salesman polytope", *Mathematical Programming*, **8** (1975), 378 - 381.

[45] M. Grötschel and M. W. Padberg, "Lineare Charakterisierungen von Travelling Salesman Problemen", *Zeitschrift für Operations Research*, **21** (1977), 33 - 64.

[46] Martin Grötschel and M. W. Padberg, "Polyhedral Theory", in *The Traveling Salesman Problem*, edited by E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys, John Wiley & Sons, 1985.

[47] M. Grötschel and W. R. Pulleyblank, "Clique Tree Inequalities and the Symmetric Traveling Salesman Problem", *Mathematics of Operations Research*, **11** (1986), 537 - 569.

[48] Michael Held and Richard M. Karp, "The traveling-salesman problem and minimum spanning trees", *Operations Research*, **18** (1970), 1138 - 1162.

[49] Michael Held and Richard M. Karp, "The traveling-salesman problem and minimum spanning trees: part II", *Mathematical Programming*, **1** (1971), 6 - 25.

[50] Harry T. Hsu, "An Algorithm for Finding a Minimum Equivalent Graph of a Digraph", *Journal of the Association for Computing Machinery*, **22** (1975), 11 - 16.

[51] D. Johnson, G. Gutin, L. McGeoch, A. Yeo, W. Zhang and A. Zverovitch, "Experimental analysis of heuristics for the ATSP", in *The Traveling Salesman Problem and its Variations*, edited by G. Gutin and A. Punnen, Springer, 2007.

[52] Haim Kaplan, Moshe Lewenstein, Nira Shafrir, and Maxim Sviridenko, "Approximation Algorithms for Asymmetric TSP by Decomposing Directed Regular Multigraphs", *Journal of the ACM*, **52** (2005), 602 - 626.

[53] R. M. Karp, "Reducibility among combinatorial problems", in *Complexity of Computer Computations* (R. E. Miller and J. W. Thatcher, eds.), Plenum Press, New York, 1972, 85 - 103.

[54] R. M. Karp and J. M. Steele, "Probabilistic analysis of heuristics", in *The Traveling Salesman Problem*, edited by E. L. Lawler, J. K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys, John Wiley & Sons, 1985.

[55] L. G. Khachiyan, "Polynomial algorithms in linear programming", *U.S.S.R. Computational Mathematics and Mathematical Physics*, **20** (1980), 53 - 72.

[56] Victor Klee and George J. Minty, "How good is the simplex algorithm?", in *Inequalities III*, ed. O. Shisha, Academic Press, 1972, 159 - 175.

[57] A. H. Land and A. G. Doig, "An automatic method of solving discrete programming problems", *Econometrica*, **28** (1960), 497 - 520.

[58] E. L. Lawler, J, K. Lenstra, A. H. G. Rinnooy Kan, and D. B. Shmoys (eds.), *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*, John Wiley & Sons, New York, 1985.

[59] Timo Leipälä and Olli Nevalainen, "Optimization of the movements of a component placement machine", *European Journal of Operational Research*, **38** (1989), 167 - 177.

[60] J. D. C. Little, K. G. Murty, D. W. Sweeney, C. Karel, "An algorithm for the traveling salesman problem", *Operations Research*, **11** (1963), 972 - 989.

[61] F. Maldonado, A. Ciurlizza, R. Radillo, and E. Ponce de León, "Optimisation of the colour sequence in the dyeing process: industrial applications", *Coloration Technology*, **116** (2000), 359 - 362.

[62] Brendan D. McKay, "Isomorph-free exhaustive generation", *Journal of Algorithms*, **26** (1998), 306 - 324.

[63] Brendan McKay, *NAUTY*, version 2.2, http://cs.anu.edu.au/~bdm/nauty/, 2003.

[64] Vardeges Melkonian and Éva Tardos, "Primal-Dual-Based Algorithms for a Directed Network Design Problem", *INFORMS Journal on Computing*, **17** (2005), 159 - 174.

[65] H. Minkowski, *Gesammelte Abhandlungen*, Leipzig: Teubner, 1911.

[66] Clyde L. Monma, Beth Spellman Munson, and William R. Pulleyblank, "Minimum-weight two-connected spanning networks", *Mathematical Programming*, **46** (1990), 153 - 171.

[67] T. S. Motzkin, H. Raiffa, G. L. Thompson, and R. M. Thrall, "The double description method" in *Contributions of the Theory of Games, Volume 2*, H. W. Kuhn and A. W. Tucker, editors, Annals of Mathematics, **8** (1953), Princeton University Press, Princeton, New Jersey, 51 - 73.

[68] Dennis M. Moyles and Gerald L. Thompson, "An Algorithm for Finding a Minimum Equivalent Graph of a Digraph", *Journal of the Association for Computing Machinery*, **16** (1969), 455 - 460.

[69] K. G. Murty, "An algorithm for ranking all the assignments in order of increasing cost", *Operations Research*, **16** (1968), 682 - 687.

[70] Denis Naddef and Giovanni Rinaldi, "The graphical relaxation: A new framework for the Symmetric Traveling Salesman Polytope", *Mathematical Programming*, **58** (1992), 53 - 88.

[71] George L. Nemhauser and Laurence A. Wolsey, *Integer and combinatorial optimization*, John Wiley & Sons, New York, 1999.

[72] M. Padberg and G. Rinaldi, "Optimization of a 532-city symmetric traveling salesman problem by branch-and-cut", *Operations Research Letters*, **6** (1987), 1 - 7.

[73] Christos H. Papadimitriou and Santosh Vempala, "On the Approximability of the Traveling Salesman Problem", *Combinatorica*, **26** (2006), 101 - 120.

[74] Sartaj Sahni and Teofilo Gonzalez, "P-Complete Approximation Problems", *Journal of the Association for Computing Machinery*, **23** (1976), 555 - 565.

[75] Alexander Schrijver, *Theory of linear and integer programming*, John Wiley & Sons, New York, 1986.

[76] Alexander Schrijver, "On the history of combinatorial optimization (till 1960)", in *Handbook of Discrete Optimization* (K. Aardal, G. L. Nemhauser, R. Weismantel, eds.), Elsevier, Amsterdam, 2005, 1 - 68.

[77] David B. Shmoys and David P. Williamson, "Analyzing the Held-Karp TSP bound: A monotonicity property with application", *Information Processing Letters*, **35** (1990), 281 - 285.

[78] Oswald Veblen, "An Application of Modular Equations in Analysis Situs", *The Annals of Mathematics, 2nd Series*, **14** (1912), 86 - 94.

[79] Santosh Vempala and Mihalis Yannakakis, "A Convex Relaxation for the Asymmetric TSP", *Proceedings of the* 10^{th} *annual ACM - SIAM Symposium on Discrete Algorithms*, (1999), 975 - 976.

[80] Hassler Whitney, "Non-separable and planar graphs", *Transactions of the American Math Society*, **34** (1932), 339 - 362.

[81] David P. Williamson, "Analysis of the Held-Karp heuristic of the traveling salesman problem", M.S. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, 1990.

[82] L. A. Wolsey, "Heuristic analysis, linear programming and branch and bound", *Mathematical Programming Study*, **13** (1980), 121 - 134.

[83] Cliff Young, David S. Johnson, David R. Karger, and Michael D. Smith, "Near-optimal Intraprocedural Branch Alignment", *ACM SIGPLAN Notices*, **32** (1997), 183 - 193.