Combinatorial Optimization Problems with Concave Costs

by

Dan Stratila

Diploma, Computer Science, 2002
Moldova State University

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
FEB 27 2009
LIBRARIES

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in Operations Research

at the

Massachusetts Institute of Technology

February 2009

© Massachusetts Institute of Technology 2009.
All rights reserved.

Author .....
Sloan School of Management
September 12, 2008

Certified by .....
Thomas L. Magnanti
Institute Professor
Thesis Supervisor

Accepted by .....
Dimitris J. Bertsimas
Professor of Operations Research
Co-director, Operations Research Center

ARCHIVES

Combinatorial Optimization Problems with Concave Costs

by
Dan Stratila

Submitted to the Sloan School of Management
on September 12, 2008 in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Operations Research

Abstract

In the first part, we study the problem of minimizing a separable concave function over a polyhedron. We assume the concave functions are nonnegative nondecreasing on $\mathbb{R}_+$, and the polyhedron is in $\mathbb{R}^n_+$ (these assumptions can be relaxed further under suitable technical conditions). We show how to approximate this problem to $1+\epsilon$ precision in optimal value by a piecewise linear minimization problem so that the number of resulting pieces is polynomial in the input size of the original problem and linear in $1/\epsilon$. For several concave cost problems, the resulting piecewise linear problem can be reformulated as a classical combinatorial optimization problem. As a result of our bound, a variety of polynomial-time heuristics, approximation algorithms, and exact algorithms for classical combinatorial optimization problems immediately yield polynomial-time heuristics, approximation algorithms, and fully polynomial-time approximation schemes for the corresponding concave cost problems. For example, we obtain a new approximation algorithm for concave cost facility location, and a new heuristic for concave cost multicommodity flow.

In the second part, we study several concave cost problems and the corresponding combinatorial optimization problems. We develop an algorithm design technique that yields a strongly polynomial primal-dual algorithm for a concave cost problem whenever such an algorithm exists for the corresponding combinatorial optimization problem. Our technique preserves constant-factor approximation ratios as well as ratios that depend only on certain problem parameters, and exact algorithms yield exact algorithms. For example, we obtain new approximation algorithms for concave cost facility location and concave cost joint replenishment, and a new exact algorithm for concave cost lot-sizing.

In the third part, we study a real-time optimization problem arising in the operations of a leading internet retailer. The problem involves the assignment of orders that arrive via the retailer's website to the retailer's warehouses. We model it as a concave cost facility location problem, and employ existing primal-dual algorithms and approximations of concave cost functions to solve it. On past data, we obtain solutions on average within 1.5% of optimality, with running times of less than 100ms per problem.

Thesis Supervisor: Thomas L. Magnanti
Title: Institute Professor

Acknowledgments

I am grateful to my advisor, Tom Magnanti, for his support and guidance over the years. Thanks to him, I have learned a lot about research and teaching during my graduate school years. Tom's encouragement at every step of the way has been invaluable, and I very much appreciated his willingness to experiment and let me pursue my own ideas. Inevitably, this meant that I have also learned a lot from my mistakes while being his student.

I would like to thank the other members of my committee, Michel Goemans, Retsef Levi, and David Simchi-Levi for their insightful comments and ideas on the research in this thesis and beyond. I am grateful to Jim Orlin for his advice and guidance ever since I entered graduate school. I learned in many ways from Jim—from refereeing papers to being a teaching assistant. I thank Russell Allgor for his input and support of my computational research. I would also like to thank Dimitris Bertsimas and Georgia Perakis for their help and advice.

I was fortunate to have started at MIT at a time when Martin Skutella was a visiting professor and was teaching the basic integer programming and combinatorial optimization course—I have never learned more per unit of time in any other class. In subsequent years, Michel Goemans' advanced graduate classes on combinatorial optimization and polytopes have been a very enriching experience.

Thanks to my fellow ORC students, and especially Raghavendran Sivaraman, Pranava Goundan, and Ping Xu, as well as my officemates over the years, Alexandre Belloni, Shobhit Gupta, Kwong Meng Teo, and Pavithra Harsha. Many thanks to Alex Andoni and Dumitru Daniliuc. I am very grateful to Carol Meyers, among other things for her unbounded patience and always very positive attitude.

My thanks go to my host family at MIT, Ellen and Richard Lacroix.

Finally, I would like to thank my parents and grandparents.

This research was supported in part by the Air Force Office of Scientific Research, Amazon.com, and the Singapore-MIT Alliance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this thesis, we study problems that can be written as the minimization of a concave cost function over a polyhedron. Such problems arise frequently in fields such as transportation, logistics, telecommunications, and supply chain management. In a typical application, the polyhedral ground set is due to network structure, capacity requirements, and other constraints, while the concave costs are due to economies of scale, volume discounts, and other practical factors [see e.g. GP90]. In particular, we obtain new results for concave cost facility location, concave cost multicommodity flow, concave cost lot-sizing, and concave cost joint replenishment.

We are interested in two types of solution methods. First, we seek algorithms with provable theoretical properties—for example, exact polynomial-time algorithms, polynomial-time approximation schemes, and approximaiton algorithms. Second, we are interested in algorithms that do not have theoretical properties or have properties that are unattractive in practice, but that nonetheless can be shown to perform well in computational experiments to an extent that makes them promising candidates for practical implementation.

The methods for solving separable concave cost minimization problems with polyhedral feasible sets can be classified into two broad categories. The methods in the first category rely on piecewise-linear approximations. After one approximates the concave cost functions by piecewise-linear functions, the resulting problem can be written as a mixed-integer program, and often this mixed-integer program represents a classical combinatorial optimization problem. Therefore, a variety of methods from integer programming and combinatorial optimization become immediately available for solving the concave cost problem.

In order for this approach to be successful, we must be able to approximate the concave cost problem by a *single* piecewise-linear problem that has few pieces and at the same time provides a good approximation to the original problem in terms of optimal cost. Chapter 2 focuses on devising methods for approximating concave functions and on bounding the resulting number of pieces.

The second broad category for solving concave cost problems consists of algorithms designed to operate directly on concave cost problems. In this category we include algorithms that perform piecewise-linear approximations iteratively as part of their computations. Di-

rect algorithms can avoid the quality of approximation/number of pieces tradeoff inherent in any piecewise-linear approach, and can overcome several other limitations of the piecewise-linear approach as well.

However, direct algorithms must be designed with concave cost problems in mind, often on an individual problem basis. For this reason, there are significantly fewer algorithms for concave cost problems than for the corresponding (or comparable) classical combinatorial optimization problems. In Chapter 3 of this thesis, we develop a technique for obtaining primal-dual algorithms for concave cost problems based on such algorithms for combinatorial optimization problems. First we develop several general insights, and then apply them to three specific problems.

In the remainder of this chapter, we introduce the specific problems studied in this thesis, describe their basic properties, and review the literature on them. Section 1.1 describes the concave cost facility location problem, and Section 1.2 the concave cost multicommodity flow problem. Then we introduce two inventory problems—concave cost lot-sizing in Section 1.3 and concave cost joint replenishment in Section 1.4.

In Chapter 2 we develop new methods and bounds for piecewise-linear approximations of concave functions. In Section 2.1 we conduct a literature reivew, and in Sections 2.2 and 2.3 we develop a general technique for concave cost problems. In Sections 2.4 and 2.5 we apply the technique to concave cost multicommodity flow and concave cost facility location.

In Chapter 3 we develop a technique for obtaining primal-dual algorithms for concave cost problems. Section 3.1 contains a literature review, Section 3.2 introduces some preliminary notions, Section 3.3 develops our technique on the basis of facility location, and Sections 3.4 and 3.5 apply it to lot-sizing and joint replenishment.

In Chapter 4 we develop a computational solution method for an order assignment problem encountered at a large internet retailer. We describe the problem in Section 4.1, model it as a concave cost facility location problem in Section 4.2, and present computational results in Section 4.3.

1.1 Concave Cost Facility Location

Let $[n] = \{1, \ldots, n\}$. In the *concave cost facility location* problem, there are m customers and n facilities. Each customer i has a demand $d_i \geq 0$, and needs to be connected to a facility to satisfy this demand. Connecting a customer i to a facility j incurs a connection cost $c_{ij}d_i$; we assume the connection costs are nonnegative and satisfy the metric inequality. Let $x_{ij} = 1$ if customer i is connected to facility j, and $x_{ij} = 0$ otherwise. Then the total demand satisfied by facility j is $\sum_{i=1}^{m} d_i x_{ij}$. Each facility j has an associated concave cost function $\phi_j : \mathbb{R}_+ \to \mathbb{R}_+$, and we assume the functions ϕ_j are nondecreasing. We also assume without loss of generality that $\phi_j(0) = 0$ for $j \in [n]$. At each facility j we incur a facility cost of $\phi_j\left(\sum_{i=1}^{m} d_i c_{ij}\right)$. The goal of the problem is to assign each customer to a facility, while minimizing the total connection and facility cost.

The concave cost facility location problem can be formulated as a mathematical program:

$$Z^*_{1.1} = \min \sum_{j=1}^{n} \phi_j \left(\sum_{i=1}^{m} d_i x_{ij} \right) + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} d_i x_{ij}, \tag{1.1a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \qquad i \in [m], \tag{1.1b}$$

$$x_{ij} \geq 0, \qquad i \in [m], j \in [n]. \tag{1.1c}$$

Given a solution x to this problem, we will refer to its cost as $Z_{1.1}(x)$. (Sometimes the problem is formulated to include the constraints $x_{ij} \in \{0, 1\}$. Since we are minimizing a concave function over a polyhedron, there is always an optimal solution at an extreme point of the polyhedron [e.g. Bau58], and an extreme point solution automatically satisfies these constraints.)

The *classical facility location* problem is the special case when the cost ϕ_j at each facility has the form

$$\phi_j(\xi_j) = \begin{cases} f_j, & \xi_j > 0, \\ 0, & \xi_j = 0. \end{cases} \tag{1.2}$$

Such a cost function is also known in the literature as a *fixed charge*. We will sometimes refer to the classical facility location problem as the *fixed-charge facility location* problem.

The classical facility location can be formulated as a mixed-integer program. To do so, introduce a binary variable y_j for each facility j, with the interpretation that $y_j = 1$ if j is open, and $y_j = 0$ otherwise:

$$\min \sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} d_i x_{ij}, \tag{1.3a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \qquad i \in [m], \tag{1.3b}$$

$$0 \leq x_{ij} \leq y_j, \qquad i \in [m], j \in [n], \tag{1.3c}$$

$$y_j \in \{0, 1\}, \qquad i \in [m], j \in [n]. \tag{1.3d}$$

Next, we describe two basic properties of the concave facility location problem. A concave function defined on $[0, +\infty)$ is continuous everywhere except at 0. Next, we consider concave functions that, in addition, are piecewise-linear everywhere except at 0.

Lemma 1. *Consider the concave cost facility location problem. Let the cost functions ϕ_j be piecewise-linear everywhere except at 0, and consist of P pieces each. Then the problem can be reduced to the classical facility location problem with P times as many facilities.*

Proof. We start with the concave cost problem. Since a piecewise-linear concave function

consisting of P pieces can be expressed as the minimum of P affine functions, we can write

$$\phi_j(\xi_j) = \begin{cases} \min\{f_{jp} + s_{jp}\xi_j : p \in [P]\}, & \xi_j > 0, \\ 0, & \xi_j = 0. \end{cases} \tag{1.4}$$

Substituting this equation into problem (1.1) and reformulating, we obtain:

$$\min \sum_{j=1}^{n}\sum_{p=1}^{P} f_{jp}y_{jp} + \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{p=1}^{P}(c_{ij} + s_{jp})d_i x_{ijp}, \tag{1.5a}$$

$$\text{s.t.} \sum_{j=1}^{n}\sum_{p=1}^{P} x_{ijp} = 1, \qquad i \in [m], \tag{1.5b}$$

$$0 \le x_{ijp} \le y_{jp}, \qquad i \in [m], j \in [n], p \in [P], \tag{1.5c}$$

$$y_{jp} \in \{0,1\}, \qquad i \in [m], j \in [n], p \in [P]. \tag{1.5d}$$

This mixed-integer program has the structure of a classical facility location problem with Pn facilities and m customers. Every piece p in the cost function ϕ_j of every facility j in the original concave cost problem corresponds to a facility $\{j,p\}$ in the new problem. The new facility has opening cost f_{jp}. The customer set is the same, and the connection cost from facility $\{j,p\}$ to customer i is $c_{ij} + s_{jp}$.

The new facility costs f_{jp} and the new connection costs $c_{ij} + s_{jp}$ are nonnegative. Since the original connection costs c_{ij} satisfy the metric inequality, so do the new connection costs $c_{ij} + s_{jp}$. Therefore, formulation (1.5) is a classical facility location problem. □

This reduction is well known in the literature, and dates back to the 1960-s [e.g. FLR66]. More recently this reduction has been employed for example by Hajiaghayi et al. [HMM03] and Mahdian et al. [MYZ06].

When the concave functions are piecewise-linear and the pieces are given explicitly in the input, the size of the resulting classical facility location problem is polynomial in the size of the original concave cost problem. Therefore, a polynomial-time approximation algorithm for the classical facility location problem immediately yields a polynomial-time approximation algorithm for the concave cost facility location problem. The same is true for polynomial-time heuristics.

However, this reduction does not result in a polynomially sized instance when the functions are given by an oracle, or are nonlinear and given algebraically. And computationally, when each function consists of a large number of pieces, the resulting algorithms can be very inefficient.

Next, we show how to use the demand structure to perform this reduction for general concave functions. Let $X = \left\{\sum_{i\in S} d_i : S \subseteq [m], S \neq \emptyset\right\}$. Since we are minimizing a concave function over a polyhedron, the concave cost problem always has an optimal solution x^* such that for every facility j, the total demand of all customers assigned to j is in X, i.e. $\sum_{i=1}^{m} d_i x^*_{ij} \in X$.

Lemma 2. *The concave cost facility location problem can be reduced to the classical facility location problem with $|X|$ times as many facilities.*

Proof. Take concave cost problem (1.1). We approximate each concave function ϕ_j by a concave function ψ_j that is piecewise-linear everywhere except at 0, consists of $|X|$ pieces, and is such that $\psi_j(\xi_j) = \phi_j(\xi_j)$ for $\xi_j \in X$, and $\psi_j(\xi_j) \geq \phi_j(\xi_j)$ for $\xi_j \geq 0$. This can be done, for example, by taking a supergradient $f_{jp} + s_{jp}\xi_j$ to ϕ_j at each point $p \in X$, and then letting $\psi_j(\xi_j) = \min\{f_{jp} + s_{jp}\xi_j : p \in X\}$ for $\xi_j > 0$, and $\psi_j(0) = 0$.

Now consider the concave cost problem obtained from problem (1.1) by replacing the functions ϕ_j with ψ_j:

$$Z^*_{1.6} = \min \sum_{j=1}^{n} \psi_j \left(\sum_{i=1}^{m} d_i x_{ij} \right) + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} d_i x_{ij}, \tag{1.6a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \qquad i \in [m], \tag{1.6b}$$

$$x_{ij} \geq 0, \qquad i \in [m], j \in [n]. \tag{1.6c}$$

Let x^* be an optimal solution to problem (1.1) with $\sum_{i=1}^{m} d_i x^*_{ij} \in X$ for $j \in [n]$. Since $\psi_j(\xi_j) = \phi_j(\xi_j)$ for $\xi_j \in X$, we have $Z^*_{1.6} \leq Z_{1.6}(x^*) = Z_{1.1}(x^*) = Z^*_{1.1}$. Conversely, let y^* be an optimal solution to problem (1.6). Since $\psi_j(\xi_j) \geq \phi_j(\xi_j)$ for $\xi_j \geq 0$, we have $Z^*_{1.6} = Z_{1.6}(y^*) \geq Z_{1.1}(y^*) \geq Z^*_{1.1}$. Therefore, $Z^*_{1.6} = Z^*_{1.1}$.

By Lemma 1, problem (1.6) can be reduced to the classical facility location problem with $|X|$ times as many facilities. □

When the concave cost facility location has uniform demands, i.e. $d_1 = d_2 = \cdots = d_m$, we have $|X| = m$, and this reduction yields a classical facility location problem with m times as many facilities. Therefore, in this case too, a polynomial-time approximation algorithm or heuristic for the classical facility location problem immediately yields a polynomial-time approximation algorithm or heuristic for the concave cost facility location problem.

In general, $|X|$ is exponentially sized, and this approach does not lead to polynomial-time algorithms for the concave cost problem.

1.1.1 Literature Review

The classical facility location problem is one of the fundamental problems in operations research, and has been studied since at least the 1960's [e.g. KH63, Sto63, Bal66]. The reference book edited by Mirchandani and Francis [MF90] introduces and reviews the literature for many problems, including the uncapacitated facility location problem [CNW90]. Since in this thesis, our main contributions to facility location problems are in the area of approximation algorithms, we next provide a survey of previous approximation algorithms for classical facility location.

The first approximation algorithm for this problem was proposed by Hochbaum [Hoc82] and applied to the case when the connection costs c_{ij} did not necessarily satisfy the metric inequality. This algorithm achieved an approximation ratio of $O(\log n)$. Shmoys et al. [SETA97] gave the first constant-factor approximation algorithm for this problem.

More recently, Jain et al. introduced 1.861 and 1.61 primal-dual approximation algorithms [JMM+03]. Sviridenko [Svi02] obtained a 1.582 approximation algorithm based on LP rounding. Mahdian et al [MYZ06] developed a 1.52 approximation algorithm that combines a primal-dual stage with a scaling stage. Currently, the best known ratio is 1.4991, achieved by an algorithm that employs a combination of LP rounding and primal-dual techniques, due to Byrka [Byr07].

Concerning the complexity of approximation, the problem where the connection costs do not necessarily obey the metric inequality has the set cover problem as a special case, and therefore is not approximable to within a constant factor unless P=NP [e.g. RS97]. The problem with metric costs is not approximable to within a factor of 1.46 unless NP $\subseteq$ DTIME $\left(n^{O(\log \log n)}\right)$ [GK99].

A central feature of location models is the economies of scale that can be achieved by connecting multiple customers to the same facility. The classical facility location problem models this effect by assigning to each facility a fixed-charge cost, as defined in equation (1.2). As one of the simplest forms of concave functions, fixed-charge costs enable the model to capture the essential tradeoff between opening many facilities in order to decrease the connection costs and opening few facilities to decrease the facility costs. The concave cost facility location problem generalizes this model by assigning to each facility a general concave cost function, and as such captures a much wider variety of phenomena than is possible with fixed costs alone.

The concave cost facility location problem has also been studied since at least the 1960's [e.g. KH63, FLR66]. Since it contains classical facility location as a special case, the previously mentioned lower bounds on approximability hold for this problem—the nonmetric version cannot be approximated within a constant factor unless P=NP, and the metric problem cannot be approximated to within a factor of 1.46 unless NP $\subseteq$ DTIME $\left(n^{O(\log \log n)}\right)$.

However, the results in the area of approximation algorithms are significantly more limited than for the classical problem. To the best of our knowledge, previously the only constant factor approximation algorithm was obtained by Mahdian and Pal [MP03]. They developed a $3 + \epsilon$ approximation algorithm based on local search. Their analysis is based in part on the analysis of the local search algorithm by Charikar and Guha [CG99, CG05].

When the concave cost facility location problem has uniform demands, a wider variety of results become available. Hajiaghayi et al. [HMM03] obtained a 1.861 approximation algorithm. A number of results become available due to the fact that, as described in Lemma 2, concave cost facility location with uniform demands can be reduced in polynomial size to classical facility location. For example, Hajiaghayi et al. [HMM03] and Mahdian et al. [MYZ06] described a 1.52 approximation algorithm.

After completion of this research, we learned of the independent research of Romeijn et al. [RSSZ07]. They develop 1.61 and 1.52 approximation algorithms for concave cost

facility location, by considering the corresponding algorithms for classical facility location [JMM+03, MYZ06] through a greedy perspective. Since our analysis in Chapter 3 uses a primal-dual perspective, establishing a connection between the research of Romeijn et al. and ours is an interesting question.

1.2 Concave Cost Multicommodity Flow

The *concave cost multicommodity flow* problem is defined on an undirected network (V, E) with node set V and edge set E; we let $n = |V|$ and $m = |E|$. On this network, there flow K commodities, and each commodity k has sources and sinks, which are given via a demand and supply vector $b_i^k, i \in V$. A coefficient $b_i^k > 0$ indicates that node i is a source of commodity k, and the supply is b_i^k; $b_i^k < 0$ indicates that i is a sink of commodity k and the demand is $-b_i^k$. We assume that the supply and demand are balanced, that is $\sum_{i \in V} b_i^k = 0$ for $k \in [K]$, and that the network is connected, i.e. it contains a path between any two nodes.

Each edge $\{i, j\} \in E$ has an associated concave cost function $\phi_{ij} : \mathbb{R}_+ \to \mathbb{R}_+$, and we assume that ϕ_{ij} are nondecreasing. Without loss of generality, we let $\phi_{ij}(0) = 0$ for $\{i, j\} \in E$. For an edge $\{i, j\} \in E$, let x_{ij}^k indicate the flow of commodity k from i to j, and x_{ji}^k the flow in the opposite direction. Then the total cost of routing flow on edge $\{i, j\}$ is $\phi_{ij}\left(\sum_{k=1}^K (x_{ij}^k + x_{ji}^k)\right)$. The goal is to route the flow of each commodity so as to satisfy the demand at all sinks, while minimizing total routing cost.

A mathematical programming formulation for this problem is given by:

$$\min \sum_{\{i,j\} \in E} \phi_{ij} \left(\sum_{k=1}^{K} (x_{ij}^k + x_{ji}^k) \right), \tag{1.7a}$$

$$\text{s.t.} \sum_{\{i,j\} \in E} x_{ij}^k - \sum_{\{j,i\} \in E} x_{ji}^k = b_i^k, \qquad i \in V, k \in [K], \tag{1.7b}$$

$$x_{ij}^k, x_{ji}^k \geq 0, \qquad \{i, j\} \in E, k \in [K]. \tag{1.7c}$$

Let $\xi_{ij} = \sum_{k=1}^K (x_{ij}^k + x_{ji}^k)$ be the total flow on an edge $\{i, j\} \in E$. The *fixed-charge multicommodity flow* problem is the special case when the edge cost functions have the form:

$$\phi_{ij}(\xi_{ij}) = \begin{cases} f_{ij} + s_{ij}\xi_{ij}, & \xi_{ij} > 0, \\ 0, & \xi_{ij} = 0. \end{cases} \tag{1.8}$$

Similarly to the functions (1.2), these functions are often referred to as fixed-charge functions in the literature. Let $B^k = \sum_{i:b_i^k > 0} b_i^k$ be the total supply of commodity k, and $B = \sum_{k=1}^K B^k$ be the total supply of all commodities.

The fixed-charge multicommodity flow problem can be formulated as a mixed integer program:

$$\min \sum_{\{i,j\}\in E} f_{ij}y_{ij} + \sum_{\{i,j\}\in E}\sum_{k=1}^{K} c_{ij}(x_{ij}^k + x_{ji}^k), \tag{1.9a}$$

$$\text{s.t.} \sum_{\{i,j\}\in E} x_{ij}^k - \sum_{\{j,i\}\in E} x_{ji}^k = b_i^k, \qquad i \in V, k \in [K], \tag{1.9b}$$

$$0 \le x_{ij}^k, x_{ji}^k \le B^k y_{ij}, \qquad \{i,j\} \in E, k \in [K], \tag{1.9c}$$

$$y_{ij} \in \{0,1\}, \qquad \{i,j\} \in E. \tag{1.9d}$$

In this setting, for each edge $\{i,j\}$, the coefficient f_{ij} can be interpreted as its installation cost, and c_{ij} as the per-unit cost of routing flow on the edge once installed.

The fixed-charge multicommodity flow problem when each commodity has only one source or one sink can be reduced to the case when each commodity has one source *and* one sink, by creating a new commodity for each source-sink pair. The latter problem is also called *uncapacitated network design* in the literature [e.g. BMW89]. Our theoretical results in Chapter 2 will apply to concave cost multicommodity flows with any demand pattern, while our computational results in Section 2.4 will focus on concave cost multicommodity flow where each commodity has one source and one sink.

As in the case of facility location, when the concave cost functions are piecewise linear everywhere except at 0 and consist of P pieces each, the concave cost multicommodity flow problem reduces to a fixed-charge multicommodity flow problem with P times as many edges. Let the concave functions be:

$$\phi_{ij}(\xi_{ij}) = \begin{cases} \min\{f_{ijp} + s_{ijp}\xi_{ij} : p \in [P]\}, & \xi_{ij} > 0, \\ 0, & \xi_{ij} = 0. \end{cases} \tag{1.10}$$

Substituting these expressions into problem (1.1) and reformulating, we obtain:

$$\min \sum_{\{i,j,p\}\in E} f_{ijp}y_{ijp} + \sum_{\{i,j,p\}\in E}\sum_{k=1}^{K} s_{ijp}(x_{ijp}^k + x_{jip}^k)), \tag{1.11a}$$

$$\text{s.t.} \sum_{\{i,j,p\}\in E} x_{ijp}^k - \sum_{\{j,i,p\}\in E} x_{ijp}^k = b_i^k, \qquad i \in V, k \in [K], \tag{1.11b}$$

$$0 \le x_{ijp}^k, x_{jip}^k \le B^k y_{ijp}, \qquad \{i,j,p\} \in E, k \in [K], \tag{1.11c}$$

$$y_{ijp} \in \{0,1\}, \qquad \{i,j,p\} \in E. \tag{1.11d}$$

This formulation is a fixed-charge multicommodity flow problem with Pm edges. For each edge $\{i,j\}$ in the original problem, the new problem has P parallel edges between nodes i and j, and $\{i,j,p\}$ refers to an edge between nodes i and j, with p being an index that distinguishes parallel edges.

Lemma 3. *Consider the concave cost multicommodity flow problem with cost functions that are piecewise-linear everywhere except at* 0*, and consist of* P *pieces each. This problem can be reduced to a fixed-charge multicommodity flow problem with* P *times as many edges.*

This reduction has the same theoretical and computational limitations as those described for concave cost facility location.

We can also use the demand structure to reduce a problem with concave functions to a piecewise-linear problem, and then reduce further to a fixed-charge multicommodity flow problem. However, the set X has to be defined differently than in Section 1.1. A simple way to do so is to take the greatest common divisor. Let $b_{\min} = \text{GCD}\{b_i^k : b_i^k \neq 0, k \in [K], i \in V\}$, and then let $X = \{b_{\min}, 2b_{\min}, \ldots, \lceil B/b_{\min} \rceil\}$.

Lemma 4. *The concave cost multicommodity flow problem can be reduced to the fixed-charge multicommodity flow problem with* $|X|$ *times as many edges.*

As with facility location, this reduction yields polynomially-sized instances when the demands are uniform, but not in general.

1.2.1 Literature Review

The fixed-charge multicommodity flow is also a fundamental problem in optimization, and has applications in communication networks, transportation, logistics, and supply chain management [see e.g. MW84, BMMN95]. Guisewite and Pardalos [GP90] survey applications and solution methods for this problem in the wider context of network flow problems with piecewise-linear concave costs or concave cost functions. In this thesis, we develop a computational procedure for a variant of the concave cost multicommodity flow problem that produces near-optimal solutions in our computational experiments. Therefore we focus our literature review on algorithms and heuristics that produce optimal or near-optimal solutions in computational experiments.

A variety of solution methods have been developed for this problem or its special cases. For example, Balakrishnan et al. [BMW89] developed primal-dual algorithms for the uncapacitated network design problem. Holmberg and Hellstrand [HH98] developed a solution method that combines a Lagrangian heuristic with branch-and-bound, also for the uncapacitated network design problem. More recently, Atamtürk developed new facets for this problem [Ata01], and Ortega and Wolsey [OW03] developed a branch-and-cut algorithm for the single-commodity version.

Concerning hardness of approximability, Andrews [And04] has shown that, for any constant γ, the uncapacitated network design problem cannot be approximated within a factor of $O(\log^{1/2-\gamma} n)$, unless NP $\subseteq$ ZPTIME($n^{\text{polylog}(n)}$).

The key reason why the fixed-charge multicommodity flow problem is difficult to solve is the presence of fixed costs on edges—without the fixed costs the problem decomposes into K polynomially solvable single-commodity minimum-cost flow problems. At the same time, the fixed costs are an essential feature of the model, as they reflect economies of scale in transportation networks, installation costs in network design, and other practical

phenomena. Similarly to facility location, the concave cost multicommodity flow problem generalizes this model by replacing the fixed-charge costs on edges with concave functions, and as a result is able to model a wider variety of practical behavior.

Since the concave cost multicommodity flow problem is a generalization of the fixed-charge multicommodity flow problem, it inherits the hardness of approximation bounds described above. The research on computational methods for the concave cost multicommodity flow problem is more limited. We are not aware of any computational results for large-scale concave cost multicommodity flow problems. Guisewite and Pardalos [GP90] provided a broader survey of solution methods for concave cost network flow problems.

Bell and Lamar [BL97] developed a branch-and-bound method for the concave cost single-commodity flow problem. Shectman and Sahinidis [SS98] developed a branch-and-bound method for a significantly more general problem—minimizing a separable concave function over a polyhedron—and also provided a survey of previous work. For the single-source concave cost flow problem, Fontes et al. developed a dynamic programming approach [FHC06b], and a branch-and-bound approach [FHC06a].

From a theoretical point of view, the worst-case running time of these algorithms is not bounded by polynomials. In computational experiments, researchers have been able to solve small to medium-scale problems [e.g. BL97, SS98, FHC06b, FHC06a]. However, once the problem size increases, the reported computation times suggest an exponential dependence of running time on problem size.

Another approach employed in the literature is heuristics. For example, Gallo and Sodini [GS79] developed a local search algorithm for the single-source concave cost flow problem. Guisewite and Pardalos [GP91] proposed several variants of local search for this problem, and compared them in computational experiments. Bazlamaccçi and Hindi [BH96] introduced an approach combining local search with tabu search for this problem. Also for the single-source concave cost flow problem, Fontes et al. [FHC03] developed a local search algorithm, and Fontes and Gonçalves [FG07] proposed an approach combining local search with a genetic algorithm.

The authors did not present theoretical bounds on the approximation guarantees obtained by these heuristics. With regard to computational results, Gallo and Sodini [GS79], Guisewite and Pardalos [GP91], and Bazlamaccçi and Hindi [BH96] reported significant cost improvements relative to the cost of the starting solutions for their approaches. However, they did not compare the cost of their final solutions to that of optimal solutions or to lower bounds. Fontes et al [FHC03] reported solutions on average within 0.07% of optimality, for those instances where the authors could compute optimal solutions. For larger instances the authors used a lower bound and obtained optimality gaps of less than 13.81%. The running times for larger instances suggest an exponential dependence on problem size. Fontes and Gonçalves [FG07] reported optimal solutions for those instances where the authors could provably compute the optimal solutions. However, for larger instances, the authors did not report optimality gaps. Although the running times were significantly lower than in [FHC03], they still suggest an exponential dependence on problem size.

1.3 Concave Cost Lot-Sizing

In the *concave cost lot-sizing* problem we have n discrete time intervals, and a single item (sometimes referred to as a product, or commodity). In each time interval $t \in [n]$, there is a demand $d_t \geq 0$ for the product, and this demand must be supplied from product ordered at time t, or from product ordered at a time $s < t$ and held until time t. In the inventory literature this requirement is known as no backlogging and no lost sales. The cost of placing an order for ξ_t units at time t is given by a concave function $\phi_t(\xi_t)$; we assume that $\phi_t : \mathbb{R}_+ \to \mathbb{R}_+$ is nondecreasing. In addition, we assume without loss of generality that $\phi_t(0) = 0$. Holding one unit from time t to time $t+1$ results in a cost of $h_t \geq 0$. The goal is to satisfy all demands, while minimizing the total ordering and holding cost.

For convenience, we introduce the coefficients $h_{st} = \sum_{i=s}^{t-1} h_i$. A mathematical programming formulation for the concave cost lot-sizing problem is given by:

$$\min \sum_{s=1}^{n} \phi_s \left(\sum_{t=s}^{n} d_t x_{st} \right) + \sum_{s=1}^{n} \sum_{t=s}^{n} h_{st} d_t x_{st}, \tag{1.12a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st} = 1, \qquad 1 \leq t \leq n, \tag{1.12b}$$

$$x_{st} \geq 0, \qquad 1 \leq s \leq t \leq n. \tag{1.12c}$$

The *classical lot-sizing* problem is the special case when the ordering cost functions have the fixed-charge form:

$$\phi_t(\xi_t) = \begin{cases} f_t + c_t \xi_t, & \xi_t > 0, \\ 0, & \xi_t = 0. \end{cases} \tag{1.13}$$

The coefficients f_t can then be viewed as fixed ordering costs, and the coefficients s_t as per-unit ordering costs. The problem can be formulated as a linear program:

$$\min \sum_{s=1}^{n} f_s y_s + \sum_{s=1}^{n} \sum_{t=s}^{n} (c_s + h_{st}) d_t x_{st}, \tag{1.14a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st} = 1, \qquad 1 \leq t \leq n, \tag{1.14b}$$

$$0 \leq x_{st} \leq y_s, \qquad 1 \leq s \leq t \leq n. \tag{1.14c}$$

The problem with the constraint $y_s \in \{0, 1\}$ for $s \in [n]$ is a formulation for classical lot-sizing. The fact that the constraints $y_s \in \{0, 1\}$ can be replaced with $y_s \geq 0$ is well-known [e.g. KB77].

As with facility location and multicommodity flow, when the concave ordering costs are piecewise linear with P pieces, i.e.

$$\phi_t(\xi_t) = \begin{cases} \min\{f_{tp} + c_{tp}\xi_t : p \in [P]\}, & \xi_t > 0, \\ 0, & \xi_t = 0, \end{cases} \tag{1.15}$$

we obtain the following mixed-integer program:

$$\min \sum_{s=1}^{n} \sum_{p\in[P]} f_{sp}y_{sp} + \sum_{s=1}^{n}\sum_{t=s}^{n}\sum_{p\in[P]} (c_{sp} + h_{st})d_t x_{spt}, \tag{1.16a}$$

$$\text{s.t.} \sum_{s=1}^{t} \sum_{p\in[P]} x_{spt} = 1, \qquad 1 \le t \le n, \tag{1.16b}$$

$$0 \le x_{spt} \le y_{sp}, \qquad 1 \le s \le t \le n, p \in [P], \tag{1.16c}$$

$$y_{sp} \in \{0,1\}, \qquad 1 \le s \le n, p \in [P]. \tag{1.16d}$$

This program can be viewed as a classical lot-sizing problem with P times as many periods. Each concave ordering cost is represented by P fixed-charge ordering costs in consecutive time periods, and there are no holding costs between these periods. As a result, the constraints $y_{sp} \in \{0,1\}$ can be omitted, and we obtain an LP formulation for this classical lot-sizing problem.

Lemma 5. *The lot-sizing problem with concave costs that are piecewise-linear everywhere except at 0, and consist of P pieces each can be reduced to a classical lot-sizing problem with P times as many periods.*

As is well known, the concave cost lot-sizing problem always has an optimal solution such that if there is an order at time s, it is serving a sequence of consecutive demand points $s, s+1, \ldots, t_s$ for some $t_s \le n$ [see e.g. Zan68]. This implies that in this solution, for time period s, the order quantity ξ_s is in the set $X := \{\sum_{i=s}^{t'} d_i : s \le t' \le n\}$. Note that there are only polynomially many points in this set, specifically $|X| = n - s + 1 \in O(n)$.

Therefore we can replace, without introducing an approximation error, the concave cost functions ϕ_t with piecewise-linear concave cost functions ψ_t with the property that $\psi_t(\xi_t) = \phi_t(\xi_t)$ for $\xi_t \in X$ and $\psi_t(\xi_t) \ge \phi_t(\xi_t)$ for $\xi_t \ge 0$. Each ψ_t will consist of $O(n)$ pieces. Therefore, unlike in the case of facility location and multicommodity flow, we obtain the following reduction.

Lemma 6. *The concave cost lot-sizing problem can be reduced to a classical lot-sizing problem with $O(n^2)$ periods.*

Proof. We proceed exactly as in the proof of Lemma 2 for facility location. □

Thus, any exact algorithm for classical lot-sizing immediately yields an exact algorithm for concave cost lot-sizing. However, the running time of the resulting algorithm

increases significantly—for example a $O(n \log n)$ algorithm for classical lot-sizing [FT91, AP93, WvHK92] would yield a $O(n^2 \log n)$ algorithm for concave cost lot-sizing. Therefore it is still of interest to develop specialized algorithms for concave cost lot-sizing.

1.3.1 Literature Review

The classical lot-sizing problem is one of the fundamental problems in inventory management and was introduced in the seminal papers of Manne [Man58], and Wagner and Whitin [WW58]. The literature on lot-sizing is extensive and here we provide only a brief survey of algorithmic results. Wagner and Whitin provided a $O(n^2)$ algorithm [WW58] under the assumption that $c_t \leq c_{t-1} + h_{t-1}$; this assumption is also known as the Wagner-Whitin condition, or the non-speculative condition. Zabel [Zab64], and Eppen et al [EGP69] obtained $O(n^2)$ algorithms for the general case. Federgruen and Tzur [FT91], Wagelmans et al. [WvHK92], and Aggarwal and Park [AP93] independently obtained $O(n \log n)$ algorithms for this problem.

Krarup and Bilde [KB77] showed that formulation (1.14) is integral. Levi et al. [LRS06] also showed that this formulation is integral, and gave a primal-dual algorithm to compute an optimal solution. (They do not evaluate the running time of their algorithm.) Our algorithm for concave cost lot-sizing in Section 3.4 will be based on this algorithm.

The concave cost lot-sizing problem generalizes classical lot-sizing by replacing the fixed-charge ordering costs with concave cost functions. This problem has also been studied since at least the 1960's. Wagner [Wag60] obtained an exact algorithm for this problem. Aggarwal and Park [AP93] obtain another exact algorithm with a running time $O(n^2)$.

1.4 Concave Cost Joint Replenishment

In the *concave cost joint replenishment (JRP)* problem we have n discrete time intervals, and K items (which may also be referred to as products, or commodities). For each item K, the set-up is similar to the lot-sizing problem. There is a demand $d_t^k \geq 0$ of item k in time period t, and the demand must be satisfied from an order at time t, or from inventory held from orders at times before t. There is a per-unit cost $h_t^k \geq 0$ for holding a unit of item k from time t to $t+1$. For each order of ξ_t^k units of item k at time t, we incur a cost $\phi^k(\xi_t^k)$, where $\phi^k : \mathbb{R}_+ \to \mathbb{R}_+$ is a nondecreasing concave function. Distinguishing the JRP from K separate concave cost lot-sizing problems is the fixed joint ordering cost—for each other at time t, we pay a fixed cost of $f^0 \geq 0$, independent of the number of items or units ordered at time t. Note that f^0 and ϕ^k do not depend on time.

Similarly to lot-sizing, let $h_{st}^k = \sum_{i=s}^{t-1} h_i^K$. A mathematical programming formulation for this problem is given by:

$$\min \sum_{s=1}^{n} \phi^0 \left(\sum_{t=s}^{n} \sum_{k=1}^{K} d_t^k x_{st}^k \right) + \sum_{s=1}^{n} \sum_{k=1}^{K} \phi^k \left(\sum_{t=s}^{n} d_t^k x_{st}^k \right) + \sum_{s=1}^{n} \sum_{t=s}^{n} \sum_{k=1}^{K} h_{st}^k d_t^k x_{st}^k, \tag{1.17a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st}^k = 1, \qquad 1 \leq t \leq n, k \in [K], \tag{1.17b}$$

$$x_{st}^k \geq 0, \qquad 1 \leq s \leq t \leq n, k \in [K]. \tag{1.17c}$$

To reflect the fact that the joint order cost is fixed, we take $\phi^0(\xi_t^0) = f^0$ if $\xi_t^0 > 0$, and $\phi^0(0) = 0$.

The *classical joint replenishment* problem is obtained when the item ordering cost functions ϕ^k have the form:

$$\phi^k(\xi_t^k) = \begin{cases} f^k, & \xi_t^k > 0, \\ 0, & \xi_t^k = 0. \end{cases} \tag{1.18}$$

The coefficients f^k can be viewed as per-item fixed ordering costs. The problem can be formulated as a MIP:

$$\min \sum_{s=1}^{n} f^0 y_s^0 + \sum_{s=1}^{n} \sum_{k=1}^{K} f^k y_s^k + \sum_{s=1}^{n} \sum_{t=s}^{n} \sum_{k=1}^{K} h_{st}^k d_t^k x_{st}^k, \tag{1.19a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st}^k = 1, \qquad 1 \leq t \leq n, k \in [K], \tag{1.19b}$$

$$0 \leq x_{st}^k \leq y_s^0, \qquad 1 \leq s \leq t \leq n, k \in [K], \tag{1.19c}$$

$$0 \leq x_{st}^k \leq y_s^k, \qquad 1 \leq s \leq t \leq n, k \in [K], \tag{1.19d}$$

$$y_s^0 \in \{0,1\}, y_s^k \in \{0,1\}, \qquad 1 \leq s \leq n, k \in [K]. \tag{1.19e}$$

Let us now consider the case when the item ordering cost functions ϕ^k are piecewise linear with P pieces:

$$\phi^k(\xi_t^k) = \begin{cases} \min\{f_p^k + c_p^k \xi_t^k : p \in [P]\}, & \xi_t^k > 0, \\ 0, & \xi_t^k = 0, \end{cases} \tag{1.20}$$

Unlike the cases of facility location, multicommodity flow, and lot-sizing, there are two obstacles to reducing the piecewise-linear concave cost JRP to the classical JRP. First, assume that only item 1 has piecewise-linear ordering costs. When attempting to reduce the piecewise-linear concave cost JRP, we would represent each piece p of the item ordering cost ϕ^k at time t by a new time period (t, p). This period would have a fixed item ordering cost f_p^k and a per-unit item ordering cost c_p^k. The main difficulty here is that, due to their origin, the costs f_p^k and c_p^k would vary non-monotonically over the time periods. However, in formulation (1.19) the fixed item ordering costs f^k are constant over time. The results of Levi et al. [LRS06] for the classical JRP assume that these costs are constant over time, or monotonically increasing.

Setting the cost assumption differences aside leads us to the second difficulty, which stems from the fact that there are multiple items, and different pieces of the item ordering cost functions may be employed by different items. Assuming, for simplicity, that each item ordering function consists of P pieces, we would need P^K time periods to represent each possible combination by a new time period, thereby leading to an exponentially-sized formulation.

It is possible to devise a polynomially-sized MIP for the piecewise-linear concave cost JRP, however this formulation behaves significantly worse from the viewpoint of the primal-dual algorithms that we will consider in Chapter 3. Instead, we reduce the piecewise-linear concave cost JRP to the following exponentially-sized formulation, which we call *generalized JRP*. Let $\pi = (p_1, \ldots, p_K)$.

$$\min \sum_{\substack{s\in[n] \\ \pi\in[P]^K}} f^0 y_{s\pi}^0 + \sum_{\substack{s\in[n],k\in[K] \\ \pi\in[P]^K}} f_{p_k}^k y_{s\pi}^k + \sum_{\substack{1\le s\le t\le n \\ k\in[K],\pi\in[P]^K}} (c_{p_k}^k + h_{st}^k) d_t^k x_{s\pi t}^k, \tag{1.21a}$$

$$\text{s.t.} \sum_{\substack{s\in[t] \\ \pi\in[P]^K}} x_{s\pi t}^k = 1, \qquad 1 \le t \le n, k \in [K], \tag{1.21b}$$

$$0 \le x_{s\pi t}^k \le y_{s\pi}^0, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [P]^K, \tag{1.21c}$$

$$0 \le x_{s\pi t}^k \le y_{s\pi}^k, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [P]^K, \tag{1.21d}$$

$$y_{s\pi}^0 \in \{0,1\}, y_{s\pi}^k \in \{0,1\}, \qquad 1 \le s \le n, k \in [K], \pi \in [P]^K. \tag{1.21e}$$

The intuition underlying the generalized JRP is that each time an item order is placed, there are P options for the item ordering cost. Choosing option p results in a fixed cost of f_p^k and a per-unit cost of c_p^k.

This formulation does not satisfy the cost assumptions required for the 2 approximation guarantee of the algorithm of Levi et al. [LRS06]. We will devise, starting from the algorithm of Levi et al, a 4 approximation algorithm for the generalized JRP. This algorithm will be based on the above formulation and have exponential running time. We would then employ the technique introduced in Chapter 3 to obtain a 4 approximation algorithm for the concave cost JRP, with a polynomial running time. As a byproduct, we will obtain a polynomial-time 4 approximation algorithm for the generalized JRP.

1.4.1 Literature Review

The classical joint replenishment problem is a fundamental model in inventory theory [see e.g. Jon87, AE88]. The problem is NP-hard [AJR89]. When the number of items or number of time periods is fixed, the problem can be solved in polynomial time [e.g. Zan66b, Vei69]. Federgruen and Tzur [FT94] developed a heuristic that computes $1+\epsilon$ approximate solutions provided certain input parameters are bounded. Shen et al [SSLT] obtain a $O(\log n + \log K)$ approximation algorithm for the one-warehouse multi-retailer problem, which has the JRP as a special case. Levi et al. [LRS06] provided the first constant factor approximation algorithm for the classical JRP, a 2 approximation primal-dual algorithm. Levi et al [LRS05] obtained a 2.398 approximation algorithm for the one-warehouse multi-retailer problem. Levi and Sviridenko [LS06] improved the approximation guarantee for the one-warehouse multi-retailer problem to 1.8.

The concave cost JRP generalizes the classical JRP by replacing the fixed ordering costs by concave cost functions. The methods employed by Zangwill [Zan66b] and Veinott [Vei69] can also be employed to solve the concave cost JRP in polynomial time. We are not aware of results for the concave cost JRP that go beyond those available for the classical JRP. Since prior to the work of Levi et al. [LRS06] there was not a constant factor approximation algorithm for the classical JRP, we conclude that no constant factor approximation algorithms are known for this problem.

Chapter 2

Piecewise-Linear Approximations

In this chapter we develop a general technique for optimization problems that can be written as the minimization of a separable concave function over a polyhedron. The concave functions can be nonlinear, piecewise linear with many pieces, or more generally given by an oracle. We assume that the concave functions are nonnegative and nondecreasing on $\mathbb{R}_+$, and that the polyhedron is in $\mathbb{R}^n_+$. (We can relax these assumptions further under suitable technical conditions.) In particular, the optimization problems defined in Chapter 1—concave cost facility location, concave cost multicommodity flow, concave cost lot-sizing, and concave cost joint replenishment—fit into this class of problems.

A natural approach for solving such a problem is to approximate each concave function by a piecewise-linear function, and then reformulate the resulting problem as a discrete optimization problem. Often this transformation can be carried out in a way that preserves problem structure, making it possible to apply existing discrete optimization techniques to the resulting problem. A wide variety of techniques is available for the resulting problems, including heuristics [e.g. BMW89, HH98], integer programming methods [e.g. Ata01, OW03], and approximation algorithms [e.g. JMM+03].

For this approach to be efficient, we need to be able to approximate the concave cost problem by a single piecewise-linear cost problem that meets two competing requirements. On one hand, the approximation should employ few pieces so that the resulting piecewise-linear cost problem will have small input size. On the other hand, the approximation should be precise enough that by solving the piecewise-linear cost problem we would obtain a solution to the original concave cost problem that provides an acceptable approximation in terms of optimal cost.

In this chapter, we introduce a general method for approximating a concave cost problem by a piecewise-linear cost problem that achieves a $1 + \epsilon$ approximation in terms of optimal cost, and at the same time provides a bound on the number of needed pieces that is polynomial in the input size of the concave cost problem and linear in $1/\epsilon$. Previously, no such polynomial bounds were known, even if we allow any dependence on $1/\epsilon$.

Our bound implies that polynomial-time heuristics, approximation algorithms, and exact algorithms for many discrete optimization problems immediately yield polynomial-time

heuristics, approximation algorithms, and fully polynomial-time approximation schemes for the corresponding concave cost problems. We illustrate this result by obtaining a new approximation algorithm for the concave cost facility location problem, and a new computational solution procedure for a class of large-scale concave cost multicommodity flow problems.

Under suitable technical assumptions, our method can be generalized to efficiently approximate the objective function of a maximization or minimization problem over a general feasible set, as long as the objective is nonnegative, separable, and concave. In fact, our technique is not limited to optimization problems. It is potentially applicable for approximating problems in continuous dynamic programming, continuous optimal control, algorithmic game theory, and other settings where new solution methods become available when concave functions are approximated by piecewise linear ones.

On the other hand, there are questions that involve problems representable as the minimization of a separable concave function over a polyhedron, and that cannot be fully answered using the method developed in this chapter. We will pose several such questions and they will motivate the research in Chapter 3.

2.1 Literature Review

Piecewise-linear approximations are used in a variety of contexts in science and engineering, and the literature on them is expansive [see e.g. dB01]. Here we limit ourselves to a survey of previous results on approximating separable concave functions in the context of optimization problems.

Geoffrion [Geo77] obtains several general results on approximating objective functions. One of the settings he considers is the minimization of a separable concave function over a general feasible set. He derives conditions under which a piecewise linear approximation of the objective achieves the smallest possible absolute error for a given number of pieces. He does not bound the number of pieces required to achieve a given precision.

Thakur [Tha78] considers a setting that includes the maximization of a separable concave function over a convex set defined by separable constraints. He approximates both the objective and constraint functions, and bounds the number of pieces needed to guarantee a given absolute error in terms of problem parameters and the maximum values of the first and second derivatives of the given functions.

Rosen and Pardalos [RP86] consider the minimization of a quadratic concave function over a polyhedron. They reduce the problem to a separable one, and then approximate the resulting univariate concave functions. They derive a bound on the number of pieces needed for a given precision in terms of objective function and feasible polyhedron parameters. Pardalos and Kovoor [PK90] specialize this piecewise linear technique to the minimization of a quadratic concave function over one linear constraint subject to upper and lower bounds on the variables.

Güder and Morris [GM94] study the maximization of a separable concave function over a polyhedron. They approximate the objective function and then derive bounds on the

number of pieces needed to guarantee a given absolute error in terms of objective function and feasible polyhedron parameters.

Kontogiorgis [Kon00] also studies the maximization of a separable concave function over a polyhedron. He uses techniques from numerical analysis to derive bounds on the number of pieces needed to guarantee a given absolute error, in terms of problem parameters and the maximum values of the second derivatives of the concave functions. He also presents computational results.

Each of these prior results differs from ours in that they do not provide a bound on the number of pieces, and thus the size of the resulting problem, that is polynomial in the size of the original problem, even if we allow any dependence on $1/\epsilon$.

Meyerson et al. [MMP00] remark, in the context of the single-sink concave cost multicommodity flow problem, that a "tight" approximation could be computed. Munagala [Mun03] states, in the same context, that an approximation of arbitrary precision could be obtained with a polynomial number of pieces. They do not mention specific bounds, or any details on how to do so.

Hajiaghayi et al. [HMM03] consider the unit-demand concave cost facility location problem, and employ an exact reduction by interpolating the concave functions at points $1, 2, \ldots, m$, where m is the number of customers. Mahdian et al [MYZ06] also consider the unit-demand concave cost facility location problem and employ this reduction.

2.1.1 Our Results

In Section 2.2 we introduce our piecewise-linear approximation technique, on the basis of minimization problems with general feasible sets in $\mathbb{R}^n_+$ and separable concave cost functions that are nonnegative and nondecreasing on $\mathbb{R}_+$. In this section, we assume that the problem has an optimal solution $x^* = (x_1^*, \ldots, x_n^*)$ with $x_i^* \in \{0\} \cup [l_i, u_i]$. To achieve a $1+\epsilon$ approximation, we need only $1 + \left\lceil \log_{1+4\epsilon+4\epsilon^2} \frac{u_i}{l_i} \right\rceil$ pieces for each concave component of the objective function. As $\epsilon \to 0$, the number of pieces behaves as $1 + \frac{1}{4\epsilon} \log \frac{u_i}{l_i}$. The number of pieces is the same for any concave function, and our method requires just one function evaluation per piece.

In Section 2.2.1, we show a function that requires at least $O\left(\frac{1}{\sqrt{\epsilon}} \log \frac{u_i}{l_i}\right)$ pieces to be approximated to within $1+\epsilon$ on $[l_i, u_i]$. Note that for any fixed ϵ, the number of pieces required by our approach is within a constant factor of this lower bound. On the other hand, it is an interesting open question to find out the minimum number of required pieces when ϵ is not fixed and may be arbitrarily close to zero, at least to within a constant factor. In Section 2.2.2, we describe several extensions, including to objective functions that are not monotone, and to feasible sets not contained in $\mathbb{R}^n_+$.

In Section 2.3, we obtain the main result of this chapter. We show that when the feasible set is a polyhedron, a $1+\epsilon$ approximation can be achieved with a number of pieces that is polynomial in the input size of the concave cost problem and linear in $1/\epsilon$. No additional assumptions or dependencies are necessary. We describe several generalizations, including to concave functions that are not monotone and to polyhedra not contained in

$\mathbb{R}^n_+$. In Section 2.3.1, we describe a simple way of reducing the resulting piecewise linear optimization problems to discrete optimization problems that often preserves the underlying problem structure. This reduction generalizes the reductions presented in Chapter 1 for specific problems, and has been employed in the literature before.

In Section 2.4, we illustrate our method on the concave cost multicommodity flow problem. We derive considerably smaller bounds on the number of required pieces than in the general case. Since our method preserves structure, the resulting discrete optimization problems are fixed-charge multicommodify flow problems. We perform computation experiments using a primal-dual method due to Balakrishnan et al [BMW89], and are able to solve large-scale problems with complete demand to within 4.2% of optimality, on average. The concave cost problems have up to 80 nodes, 1,580 edges, 6,320 commodities, and 9.9 million flow variables. These problems are, to the best of our knowledge, significantly larger than previously solved concave cost multicommodity flow problems with full demand. For a literature review of previous work on the concave cost multicommodity flow problem, the reader is directed to Section 1.2.

In Section 2.5, we illustrate our method on the concave cost facility location problem. Combining a 1.4991-approximation algorithm for the classical uncapacitated facility location problem due to Byrka [Byr07] with our technique, we obtain a $1.4991+\epsilon$ approximation algorithm for the concave cost facility location problem. Taking for example $\epsilon = 0.0009$, we obtain a 1.5-approximation algorithm for concave cost facility location. Previously, the approximation algorithm with the lowest ratio for this problem was a $3+\epsilon$ approximation algorithm based on local search, due to Mahdian and Pal [MP03]. Since completing this research, we have learned about the independent work of Romeijn et al [RSSZ07]. They develop 1.61 and 1.52 approximation algorithms for the concave cost facility location problem. A more detailed literature review of approximation algorithms for the concave cost facility location problem can be found in Section 1.1.

2.2 General Feasible Sets

Let $x = (x_1, \ldots, x_n)$. We examine the general concave minimization problem

$$Z^*_{2.1} = \min \{\phi(x) : x \in X\}, \tag{2.1}$$

defined by a closed feasible set $X \subseteq \mathbb{R}^n_+$, and a separable concave cost function $\phi : \mathbb{R}^n_+ \to \mathbb{R}_+$ with $\phi(x) = \sum_{i=1}^n \phi_i(x_i)$. The feasible set need not be convex or connected—for example, it could be the feasible set of an integer program. Let $[n] = \{1, \ldots, n\}$. We impose the following assumption.

Assumption 1. (a) The function ϕ is nondecreasing. (b) The problem has an optimal solution $x^* = (x^*_1, \ldots, x^*_n)$ and bounds $0 < l_i < u_i$ such that $x^*_i \in \{0\} \cup [l_i, u_i]$ for $i \in [n]$.

To approximate problem (2.1) within a factor of $1+\epsilon$, we approximate each function ϕ_i with a piecewise linear function $\psi_i : \mathbb{R}_+ \to \mathbb{R}_+$. Each function ψ_i consists of $1+P$ pieces,

with $P := \left\lceil \log_{1+\epsilon} \frac{u_i}{l_i} \right\rceil$, and is defined by the coefficients

$$s_i^p = \phi_i' \left(l_i(1+\epsilon)^p \right), \qquad p \in \{0, \ldots, P\}, \tag{2.2a}$$

$$f_i^p = \phi_i \left(l_i(1+\epsilon)^p \right) - l_i(1+\epsilon)^p s_i^p, \qquad p \in \{0, \ldots, P\}. \tag{2.2b}$$

If the derivative $\phi_i'(l_i(1+\epsilon)^p)$ does not exist, we take the derivative from the right, that is $s_i^p = \lim_{x_i \to l_i(1+\epsilon)^p +} \frac{\phi_i(p) - \phi_i(x_i)}{p - x_i}$. The derivative from the right always exists at points in $(0, +\infty)$ since ϕ_i is concave on $[0, +\infty)$.

Each coefficient pair defines a line with nonnegative slope s_i^p and y-intercept f_i^p, which is tangent to the graph of ϕ_i at the point $l_i(1+\epsilon)^p$. For $x_i > 0$, the function ψ_i is defined by the lower envelope of these lines:

$$\psi_i(x_i) = \min\{f_i^p + s_i^p x_i : p = 0, \ldots, P\}. \tag{2.3}$$

We let $\psi_i(0) = \phi_i(0)$ and $\psi(x) = \sum_{i=1}^n \psi_i(x_i)$. Substituting ψ for ϕ, we obtain the piecewise linear concave minimization problem

$$Z_{2.4}^* = \min\{\psi(x) : x \in X\}. \tag{2.4}$$

Next, we prove that this problem provides an approximation for problem (2.1). We first present a proof with an intuitive geometric interpretation, but which does not yield a tight approximation ratio. A tight analysis will follow.

Lemma 7. $Z_{2.1}^* \leq Z_{2.4}^* \leq (1+\epsilon) Z_{2.1}^*$.

Proof. Let x^* be an optimal solution of problem (2.4). The graph of any line $f_i^p + c_i^p x_i$ lies on or above the graph of $\phi_i(x_i)$, hence $\phi_i(x_i^*) \leq \psi_i(x_i^*)$ for $i \in [n]$. Therefore, $Z_{2.1}^* \leq \phi(x^*) \leq \psi(x^*) = Z_{2.4}^*$.

Conversely, let x^* be an optimal solution of problem (2.1) satisfying Assumption 1(b). It suffices to show that $\psi_i(x_i^*) \leq (1+\epsilon)\phi_i(x_i^*)$ for $i \in [n]$. If $x_i^* = 0$, then the inequality holds. Otherwise, let $p = \left\lfloor \log_{1+\epsilon} \frac{x_i^*}{l_i} \right\rfloor \geq 0$, and note that $\frac{x_i^*}{l_i} \in [(1+\epsilon)^p, (1+\epsilon)^{p+1}]$. Because ϕ_i is concave, nonnegative, and nondecreasing,

$$\psi_i(x_i^*) \leq f_i^p + s_i^p x_i^* \leq f_i^p + s_i^p l_i (1+\epsilon)^{p+1} \tag{2.5a}$$

$$= f_i^p + s_i^p l_i (1+\epsilon)(1+\epsilon)^p \leq (1+\epsilon) \left(f_i^p + s_i^p l_i (1+\epsilon)^p \right) \tag{2.5b}$$

$$= (1+\epsilon)\phi_i \left(l_i(1+\epsilon)^p \right) \leq (1+\epsilon)\phi_i(x_i^*). \tag{2.5c}$$

(See Figure 2-1 for an illustration.) Therefore, $Z_{2.4}^* \leq \psi(x^*) \leq (1+\epsilon)\phi(x^*) = (1+\epsilon) Z_{2.1}^*$. □

We now present a tight analysis, i.e. an analysis that reveals the lowest approximation ratio that is guaranteed by the approach defined by equations (2.2).

Theorem 1. $Z_{2.1}^* \leq Z_{2.4}^* \leq \frac{1+\sqrt{\epsilon+1}}{2} Z_{2.1}^* \leq (1 + \frac{\epsilon}{4}) Z_{2.1}^*$.

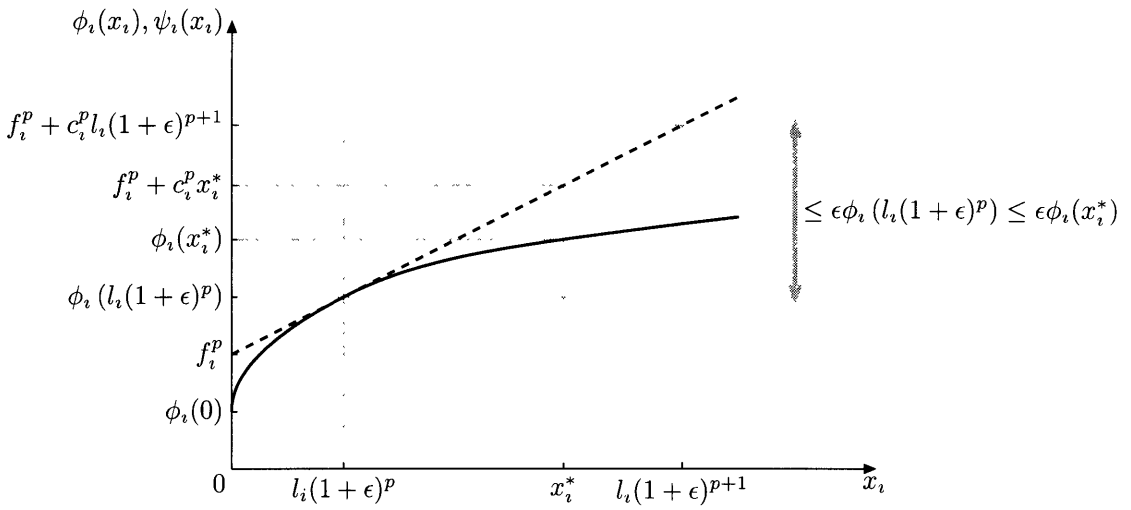

Figure 2-1: Illustration of the proof of Lemma 7. Observe that the height of any point inside the box with the bold lower left and upper right corners exceeds the height of the box's lower left corner by at most a factor of ϵ.

Proof. Without loss of generality, we assume $l_i = 1$ and $\phi_i(0) = 0$, and consider only the segment $[1, 1+\epsilon]$, and the two tangents at $(1, \phi_i(1))$ and $(1+\epsilon, \phi_i(1+\epsilon))$. Suppose these tangents have slopes a and c respectively. The worst-case approximation ratio is achievable when ϕ_i consists of 3 linear pieces with slopes $a > b > c$ on $[0,1]$, $[1, 1+\epsilon]$, and $[1+\epsilon, +\infty]$ respectively. Let $x_i^* = 1 + \xi \in [1, 1+\epsilon]$. We will now compute values of a, b, c, and ξ that yield the highest approximation ratio, as well as the ratio itself.

The values yielded by the two tangents at x_i^* are $a(1+\xi)$ and $a + b\epsilon - c(\epsilon - \xi)$, and

$$\phi_i(1+\xi) = a + b\xi, \qquad \psi_i(1+\xi) = \min\left\{a(1+\xi), a + b\epsilon - c(\epsilon - \xi)\right\}, \tag{2.6}$$

Since $\xi \leq \epsilon$, the worst case is achievable when $c = 0$. Since we seek to find ξ that maximizes

$$\frac{\psi_i(1+\xi)}{\phi_i(1+\xi)} = \min\left\{\frac{a + a\xi}{a + b\xi}, \frac{a + b\epsilon}{a + b\xi}\right\}, \tag{2.7}$$

we can assume ξ is such that $\frac{a+a\xi}{a+b\xi} = \frac{a+b\epsilon}{a+b\xi}$, which yields $\xi = \frac{b\epsilon}{a}$. Substituting, we now seek to maximize $\frac{1+\epsilon b/a}{1+\epsilon b^2/a^2}$. Since we are now seeking values of a and b that yield the highest ratio, we can substitute $d = \frac{b}{a}$, and seek a d that yields the highest ratio. The maximum is achieved at $d = \frac{-1+\sqrt{\epsilon+1}}{\epsilon}$ and equals $\frac{1+\sqrt{\epsilon+1}}{2}$, which is less than $1 + \frac{\epsilon}{4}$. □

Since $\frac{1+\sqrt{\epsilon+1}}{2} \to 1$ and $\frac{d}{d\epsilon}\frac{1+\sqrt{\epsilon+1}}{2} \to \frac{1}{4}$ as $\epsilon \to 0$, it follows that $1 + \frac{\epsilon}{4}$ is the best ratio expressible as a linear function of ϵ that can be achieved with our approach. Equivalently,

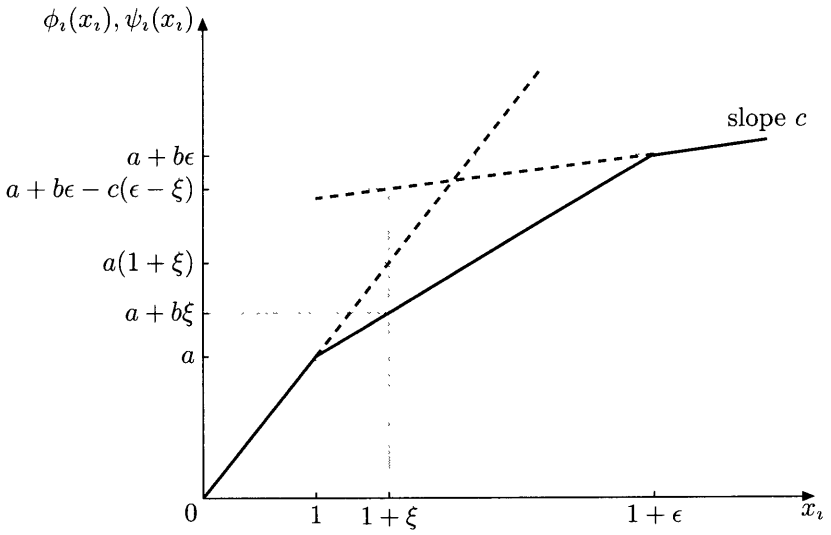

Figure 2-2: Illustration of the proof of Theorem 1.

instead of an approximation ratio of $\frac{1+\sqrt{\epsilon+1}}{2}$ using $1 + \left\lceil \log_{1+\epsilon} \frac{u_i}{l_i} \right\rceil$ pieces, we can obtain a ratio of $1+\epsilon$ using only $1 + \left\lceil \log_{1+4\epsilon+4\epsilon^2} \frac{u_i}{l_i} \right\rceil$ pieces. We can derive improved bounds on the number of pieces when the functions are known to belong to particular classes (for example, logarithmic functions), and even better bounds when the functions are known.

The number of pieces can be written as $1 + \left\lceil \frac{1}{\log(1+4\epsilon+4\epsilon^2)} \log \frac{u_i}{l_i} \right\rceil$. Note that as $\epsilon \to 0$, $\frac{1}{\log(1+4\epsilon+4\epsilon^2)} \to +\infty$ and $\frac{4\epsilon}{\log(1+4\epsilon+4\epsilon^2)} \to 1$. Therefore, as $\epsilon \to 0$, the number of pieces behaves as $1 + \frac{1}{4\epsilon} \log \frac{u_i}{l_i}$. This dependence enables us to apply the approximation technique to practical concave cost problems. In Section 2.3 we will exploit the logarithmic dependence of our results on $\frac{u_i}{l_i}$ to derive polynomial bounds on the number of pieces for a large class of problems.

2.2.1 A Lower bound on the Number of Pieces

The analysis in the proof of Theorem 1 is tight if we consider a function ϕ_i given by a, b, and c at the values obtained in the proof. Therefore, if we introduce the pieces as specified in equation (2.2), then $\frac{1+\sqrt{\epsilon+1}}{2}$ is the best approximation ratio that can be achieved.

In this section, we establish a lower bound on the number of pieces required by *any* approach. First, we show that by limiting ourselves to a piecewise-linear function that is concave and whose pieces are given by tangents, we increase the number of required pieces by at most a constant factor. Let $\phi_i(x_i)$ be a concave function, and $\psi_i(x_i)$ a piecewise linear function of Q pieces with $\frac{1}{1+\epsilon} \leq \frac{\psi_i(x_i)}{\phi_i(x_i)} \leq 1 + \epsilon$ for $x_i \in [l_i, u_i]$.

Lemma 8. *There is a piecewise-linear concave function $\varphi_i(x_i)$ consisting of at most $4Q$ pieces such that $\frac{1}{1+\epsilon} \leq \frac{\varphi_i(x_i)}{\phi_i(x_i)} \leq 1+\epsilon$ for $x_i \in [l_i, u_i]$, and each piece of φ_i is tangent to ϕ_i.*

Proof. Fix a piece of ψ_i with intercept f_i^p and slope s_i^p. We can assume that the piece is either tangent to ϕ_i, intersects it at one point, or intersects it at two points. If the piece is tangent, our argument is complete.

If the piece intersects ϕ_i at two points ξ_1 and ξ_2, it provides an $1+\epsilon$ approximation either on one interval containing ξ_1 and ξ_2, or on two intervals, one containing ξ_1 and the other ξ_2. In the case of one interval, we reduce it to the case of two consecutive intervals, touching at $\frac{\xi_1+\xi_2}{2}$. In the case of two intervals, we let the intervals be $[\xi_1', \xi_1'']$ and $[\xi_2', \xi_2'']$, with $\xi_1 \in [\xi_1', \xi_1'']$ and $\xi_2 \in [\xi_2', \xi_2'']$. We now treat this piece as two separate pieces that each intersect ϕ_i at one point, with one providing an $1+\epsilon$ approximation on $[\xi_1', \xi_1'']$ and the other on $[\xi_2', \xi_2'']$. This increases the number of needed pieces by at most a factor of 2.

It now remains to study the case when the piece interesects ϕ_i at a single point ξ, and provides an $1+\epsilon$ approximation on $[\xi', \xi'']$, with $\xi \in [\xi', \xi'']$. First, assume that the piece lays above the graph for $x_i \in [\xi', \xi)$ and below the graph for $x_i \in (\xi, \xi'']$. Since the piece is above the graph on $[\xi', \xi]$, we can guarantee an $1+\epsilon$ approximation on $[\xi', \xi]$ by introducing a tangent at ξ'. On $[\xi, \xi'']$, we can guarantee a $1+\epsilon$ approximation by introducing a piece with intercept $(1+\epsilon)f_i^p$ and slope $(1+\epsilon)s_i^p$. This piece will be above the function on $[\xi, \xi'']$, and therefore we can guarantee a $1+\epsilon$ approximation on $[\xi, \xi'']$ by introducing a tangent at ξ''. Since we have introduced two tangents for one original piece, the number of needed pieces again increases by a factor of 2, for a total factor of 4. □

Let $\phi_i(x_i) = \sqrt{x_i}$, and let ψ_i be a piecewise linear function with $\frac{1}{1+\epsilon} \leq \frac{\psi_i(x_i)}{\phi_i(x_i)} \leq 1+\epsilon$ for $x_i \in [l_i, u_i]$, and each piece of ψ_i tangent to the graph of ϕ_i. In the following lemma, we compare the number of tangents required by our approach with the minimum number of tangents needed to approximate ϕ_i.

Lemma 9. *As $\epsilon \to 0$, the minimum number of pieces in ψ_i behaves as $\frac{1}{\sqrt{32\epsilon}} \log \frac{u_i}{l_i}$. For fixed ϵ, the minimum number of pieces is within a constant factor of $1 + \left\lceil \log_{1+4\epsilon+4\epsilon^2} \frac{u_i}{l_i} \right\rceil$, the number of pieces required by our approach.*

Proof. Fix $\xi_0 \in [l, u]$ and let us determine the segment $[\xi_0(1+\delta_1), \xi_0(1+\delta_2)]$ on which a tangent to the graph of ϕ_i at ξ_0 will guarantee a $1+\epsilon$ approximation. The values of δ_1 and δ_2 are given by the solutions to the equation

$$\phi_i(\xi_0) + \delta\xi_0\phi_i'(\xi_0) = (1+\epsilon)\phi_i((1+\delta)\xi_0) \quad \Leftrightarrow \tag{2.8a}$$

$$\sqrt{\xi_0} + \delta\xi_0 \frac{1}{2\sqrt{\xi_0}} = (1+\epsilon)\sqrt{(1+\delta)\xi_0} \quad \Leftrightarrow \tag{2.8b}$$

$$\xi_0 + \delta\xi_0 + \frac{1}{4}\delta^2\xi_0 = (1+\epsilon)^2(1+\delta)\xi_0. \tag{2.8c}$$

This is simply a quadratic equation w.r.t. δ, and solving it yields $\delta_1 = 2\epsilon(2+\epsilon) - 2(1+\epsilon)\sqrt{\epsilon(2+\epsilon)}$ and $\delta_2 = 2\epsilon(2+\epsilon) + 2(1+\epsilon)\sqrt{\epsilon(2+\epsilon)}$. Let $\xi_1 = (1+\delta_1)\xi_0$. A tangent can

provide an approximation on a segment of the form

$$[\xi_1, \gamma(\epsilon)\xi_1] := \left[\xi_1, \frac{1+\delta_2}{1+\delta_1}\xi_1\right] = \left[\xi_1, \frac{1+2\epsilon(2+\epsilon)+2(1+\epsilon)\sqrt{\epsilon(2+\epsilon)}}{1+2\epsilon(2+\epsilon)-2(1+\epsilon)\sqrt{\epsilon(2+\epsilon)}}\xi_1\right]. \tag{2.9}$$

Since $\gamma(\epsilon)$ does not depend on ξ_1, it immediately follows that we need $\left\lceil \log_{\gamma(\epsilon)} \frac{u_i}{l_i} \right\rceil$ pieces to approximate ϕ_i on $[l_i, u_i]$. This is within a factor of $1 + 2\frac{\log\gamma(\epsilon)}{\log(1+4\epsilon+4\epsilon^2)}$ of the number of pieces required by our approach. Note that $\log_{\gamma(\epsilon)} \frac{u_i}{l_i} = \frac{1}{\log\gamma(\epsilon)} \log \frac{u_i}{l_i}$, and as $\epsilon \to 0$, we have $\frac{1}{\log\gamma(\epsilon)} \to +\infty$ and $\frac{\log\gamma(\epsilon)}{\sqrt{\epsilon}} \to \sqrt{32}$. Therefore, the minimum number of pieces behaves as $\frac{1}{\sqrt{32\epsilon}} \log \frac{u_i}{l_i}$ as $\epsilon \to 0$. □

If we do not restrict ourselves to tangents, by Lemma 2 the minimum number of pieces for approximating $\phi_i(x_i) = \sqrt{x_i}$ as $\epsilon \to 0$ behaves as $\frac{1}{16\sqrt{2\epsilon}} \log \frac{u_i}{l_i}$, and is within a factor of $\sqrt{\epsilon}/4\sqrt{2}$ of the number of pieces required by our approach. This implies that for fixed ϵ and any l_i and u_i, the number of pieces required by our approach is within a constant factor of the best possible. An interesting open question is to find upper and lower bounds on the number of required pieces that are tight, or tight within a constant factor as $\epsilon \to 0$.

2.2.2 Extensions

Our approach applies to a broader class of problems. Consider the problem

$$\min\{\phi(x) : x \in X\}, \tag{2.10}$$

defined by a closed feasible set $X \subseteq \mathbb{R}^n$ and a separable concave function $\phi : \mathrm{conv}(X) \to \mathbb{R}_+$. We replace Assumption 1 with the following more relaxed assumption.

Assumption 2. Problem (2.10) has an optimal solution x^* and bounds $0 < l_i < u_i$ such that $|x_i^*| \le u_i$, and either $\phi_i(x_i^*) = 0$ or $\min\{|x_i^* - x_i| : \phi_i(x_i) = 0\} \ge l$, for $i \in [n]$.

The following is a generalization of Theorem 1.

Corollary 1. *Problem (2.10) can be approximated within a factor of* $1+\epsilon$ *by replacing each function* ϕ_i *with a piecewise linear function* ψ_i *of* $2 + 2\left\lceil \log_{1+4\epsilon+4\epsilon^2} \frac{2u_i}{l_i} \right\rceil$ *pieces, and at most two discontinuity points.*

Proof. We will consider each objective component ϕ_i separately. Any concave function $\phi_i(x_i)$ that is not constant over the projection of $\mathrm{conv}(X)$ to x_i will have at most two zeroes, which we denote by $\zeta_i^{\mathrm{L}} < \zeta_i^{\mathrm{R}}$. Let $\zeta_i' = \max\{-u_i - l_i, \zeta_i^{\mathrm{L}}\}$ and $\zeta_i'' = \min\{u_i + l_i, \zeta_i^{\mathrm{R}}\}$ and note that we need to approximate ϕ_i only on $[\zeta_i' + l_i, \zeta_i'' - l_i]$. Let ζ_i^* be a point where ϕ_i is maximized, and note that ϕ_i is monotonically nondecreasing on $[\zeta_i', \zeta_i^*]$, and monotonically nonincreasing on $[\zeta_i^*, \zeta_i'']$. We will apply Theorem 1 to each of these two segments, by using translation and reflection.

If one of the two segments is empty, the proof is complete. Otherwise, w.l.o.g. consider the segment $[\zeta_i', \zeta_i^*]$. To avoid having to compute ζ_i^*, we simply introduce tangents until the slope is negative. Let the last tangent be at $\zeta_i' + l_i(1+4\epsilon+4\epsilon^2)^{P_i}$. If its slope is negative, Theorem 1 does not guarantee an approximation ratio on the segment $[\zeta_i' + l_i(1+4\epsilon+4\epsilon^2)^{P_i-1}, \zeta_i' + l_i(1+4\epsilon+4\epsilon^2)^{P_i}]$. In this case, we remove the tangent at $\zeta_i' + l_i(1+4\epsilon+4\epsilon^2)^{P_i}$, and introduce a tangent at $\zeta_i' + l_i(1+4\epsilon+4\epsilon^2)^{P_i-1}(1+\epsilon)^p$ for the largest p that yields a nonnegative slope; since $(1+\epsilon)^4 \geq 1+4\epsilon+4\epsilon^2$, $p \leq 3$. The approximation is guaranteed on $[\zeta_i' + l_i(1+4\epsilon+4\epsilon^2)^{P_i-1}, \zeta_i' + l_i(1+4\epsilon+4\epsilon^2)^{P_i-1}(1+\epsilon)^p]$ by Theorem 1, and on $[l_i(1+4\epsilon+4\epsilon^2)^{P_i-1}(1+\epsilon)^p, \xi_i^*]$ by Lemma 7.

The number of pieces employed is at most $2+\left\lceil \log_{1+4\epsilon+4\epsilon^2} \frac{\zeta_i^*-\zeta_i'}{l_i} \right\rceil + \left\lceil \log_{1+4\epsilon+4\epsilon^2} \frac{\zeta_i''-\zeta_i^*}{l_i} \right\rceil \leq 2 + 2\left\lceil \log_{1+4\epsilon+4\epsilon^2} \frac{2u_i}{l_i} \right\rceil$, since $\zeta_i'' - \zeta_i' \leq 2u_i$. Each segment yields at most one discontinuity point. □

We conclude with two further extensions:

1) We can use secants instead of tangents, in which case we require only one function evaluation per piece, and do not need to evaluate the derivative. The secant approach may be preferable in computational applications where tangents are difficult to compute.

2) The results in this section, but not in subsequent ones, also apply to concave maximization problems.

2.3 Polyhedral Feasible Sets

Let $X = \{x : Ax \leq b, x \geq 0\}$ be a rational polyhedron defined by $A \in \mathbb{Q}^{m\times n}$ and $b \in \mathbb{Q}^m$, and let $\phi : \mathbb{R}_+^n \to \mathbb{R}_+$ be a separable nondecreasing concave function. We consider the problem

$$Z_{2.11}^* = \min\{\phi(x) : Ax \leq b, x \geq 0\}. \tag{2.11}$$

We will bound the components of the optimal solution in terms of input data size. We take the input data size for problem (2.11) to be the size of A and b alone; omitting the objective functions ϕ_i from the input size computation only strengthens the resulting bounds. Following standard practice [see e.g. KV02] we define the size of rational numbers and matrices as the number of bits needed to represent them:

1) for integers $r \in \mathbb{Z}$, $\text{size}(r) := 1 + \lceil \log_2(|r|+1) \rceil$;

2) for rational numbers $r = \frac{r_1}{r_2} \in \mathbb{Q}$ with $\frac{r_1}{r_2}$ irreducible, $\text{size}(r) := \text{size}(r_1) + \text{size}(r_2)$;

3) for vectors or matrices $A \in \mathbb{Q}^{m\times n}$, $\text{size}(A) := mn + \sum_{i=1}^m \sum_{j=1}^n \text{size}(a_{ij})$.

Let $U(A,b) = 4(\text{size}(A) + \text{size}(b) + n^2 + 5n)$.The following property is well-known [see e.g. KV02, GLS93].

Lemma 10. *Any vertex x of X has* $\text{size}(x) \leq U(A,b)$.

To approximate problem (2.11), we introduce the piecewise linear functions ψ_i as described in equations (2.2) and (2.3); each function will have $1 + \left\lceil \frac{2U(A,b)}{\log_2(1+4\epsilon+4\epsilon^2)} \right\rceil$ pieces. Consider the problem

$$Z^*_{2.12} = \min\{\psi(x) : Ax \leq b, x \geq 0\}. \tag{2.12}$$

Theorem 2. $Z^*_{2.11} \leq Z^*_{2.12} \leq (1+\epsilon)Z^*_{2.11}$. *Each function ψ_i has a number of pieces polynomial in* $\text{size}(A) + \text{size}(b)$, *the input size of problem (2.11).*

Proof. Because $X \subseteq \mathbb{R}^n_+$, it has at least one vertex, and because ϕ is nonnegative, $Z_{2.11}$ is bounded from below. Therefore, because ϕ is concave, problem (2.11) has an optimal solution x^* at a vertex of X [Bau58]. Lemma 10 ensures that $x^*_i \in \{0\} \cup \left[2^{-U(A,b)}, 2^{U(A,b)}\right]$. Therefore, by Theorem 1, we can obtain the desired approximation with $1 + \left\lceil \log_{1+4\epsilon+4\epsilon^2} 2^{2U(A,b)} \right\rceil = 1 + \left\lceil \frac{2U(A,b)}{\log_2(1+4\epsilon+4\epsilon^2)} \right\rceil$ pieces. □

Again, a generalization is possible. Consider the problem

$$\min\{\phi(x) : Ax \leq b\}, \tag{2.13}$$

defined by a rational polyhedron $X = \{x : Ax \leq b\}$ with at least one vertex, and a separable concave function $\phi : X \to \mathbb{R}_+$. Any concave function $\phi_i(x_i)$ that is not constant over the projection of the feasible polyhedron to x_i will have at most two zeroes; denote them by $\zeta'_i < \zeta''_i$, and assume they are rational.

Corollary 2. *Problem (2.13) can be approximated within a factor of* $1+\epsilon$ *by replacing each function ϕ_i with a piecewise linear function ψ_i of* $2 + 2\left\lceil \frac{2U(A,b)+\text{size}(\zeta'_i)+\text{size}(\zeta''_i)+1}{\log_2(1+4\epsilon+4\epsilon^2)} \right\rceil$ *pieces, and at most two discontinuity points.*

Proof. Let x^* be a vertex optimal solution of (2.13). Then $|x^*_i| \leq u := 2^{U(A,b)}$. Moreover, either $x^*_i \in \{\zeta'_i, \zeta''_i\}$ or $\min\{|x^*_i - \zeta'_i|, |x^*_i - \zeta''_i|\} \geq l := 2^{-U(A,b)-\text{size}(\zeta'_i)-\text{size}(\zeta''_i)}$. Applying Corollary 1 completes the proof. □

This corollary is motived by the fact that in many applications ζ'_i and ζ''_i are present in the input, as part of the description of the concave cost functions. If ζ'_i and ζ''_i are part of the input, the number of pieces in each function ψ_i is polynomial in the size of the input.

2.3.1 Representing the Piecewise Linear Functions

To solve the problems resulting from our approximation technique, we could use several classical methods for representing piecewise linear functions as mixed integer programs. Such methods usually introduce one or more binary variables for each piece and add a coupling constraint that ensures the approximation uses only one piece [see e.g. NW99, CGM03]. However, since the objective function to be minimized is concave, the coupling constraint is

unnecessary, and we can employ the following well-known fixed charge formulation, which is equivalent to formulation (2.12):

$$\min \sum_{i=1}^{n} \sum_{p=1}^{P} (f_i^p z_i^p + s_i^p y_i^p), \tag{2.14a}$$

$$\text{s.t. } Ax \leq b, \tag{2.14b}$$

$$x_i = \sum_{p=0}^{P} y_i^p, \qquad i \in [n], \tag{2.14c}$$

$$0 \leq y_i^p \leq u_i z_i^p, \qquad i \in [n], p \in \{0, \ldots, P\}, \tag{2.14d}$$

$$z_i^p \in \{0, 1\}, \qquad i \in [n], p \in \{0, \ldots, P\}. \tag{2.14e}$$

We assume without loss of generality that $f(0) = 0$; if $f(0) > 0$ the approximation only becomes tighter. We choose the coefficients u_i so that $x_i \leq u_i$ at any vertex, for example $u_i = 2^{U(A,b)}$.

Lemma 11. *The input size of problem (2.14) is polynomial in the input size of problem (2.12).*

A key advantage of fixed-charge formulation (2.14) is that, in many cases, it preserves the special structure of the original concave cost problem. For example, in Chapter 1, we have seen how structure is preserved for concave cost facility location, multicommodity flow, and lot-sizing. In such cases, solution methods for fixed-charge problems can be used to approximately solve the corresponding concave cost problems.

A drawback of problem (2.14) is that it has $1 + P$ times more variables. Although for general polyhedra P could be prohibitively large, for many practical problems, we are able to derive significantly smaller bounds on P.

2.4 Multicommodity Flows

To illustrate our approach on a practical problem, we consider the concave cost multicommodity flow problem defined in Section 1.2. Recall that the problem can be formulated as follows:

$$\min \sum_{\{i,j\} \in E} \phi_{ij} \left(\sum_{k=1}^{K} (x_{ij}^k + x_{ji}^k) \right), \tag{2.15a}$$

$$\text{s.t. } \sum_{\{i,j\} \in E} x_{ij}^k - \sum_{\{j,i\} \in E} x_{ji}^k = b_i^k, \qquad i \in V, k \in [K], \tag{2.15b}$$

$$x_{ij}^k, x_{ji}^k \geq 0, \qquad \{i,j\} \in E, k \in [K]. \tag{2.15c}$$

As before, we assume that $\phi_{ij} : \mathbb{R}_+ \to \mathbb{R}_+$ are nondecreasing concave functions. The meaning of variables and coefficients is the same as in Section 1.2.

Let $B^k = \sum_{i:b_i^k>0} b_i^k$, and $B = \sum_{k=1}^K B^k$. Since rational numbers can be scaled to obtain integers, for simplicity we assume that the problem data are integral. Without loss of generality, we also assume that $\phi_{ij}(0) = 0$ for $\{i,j\} \in E$.

As shown in Section 1.2, performing a piecewise-linear approximation with Q pieces for each concave function ϕ_i, and then applying formulation (2.14) yields the well-known fixed-charge multicommodity flow problem, but now on a network with Qm edges:

$$\min \sum_{\{i,j,p\}\in E} \sum_{k=1}^{K} (f_{ijp} z_{ijp} + s_{ijp}(x_{ijp}^k + x_{jip}^k)), \tag{2.16a}$$

$$\text{s.t.} \sum_{\{i,j,p\}\in E} x_{ijp}^k - \sum_{\{j,i,p\}\in E} x_{ijp}^k = b_i^k, \qquad i \in V, k \in [K], \tag{2.16b}$$

$$0 \le x_{ijp}^k, x_{jip}^k \le B^k z_{ijp}, \qquad \{i,j,p\} \in E, k \in [K], \tag{2.16c}$$

$$z_{ijp} \in \{0,1\}, \qquad \{i,j,p\} \in E. \tag{2.16d}$$

In this formulation, there may be multiple edges between two nodes, and $\{i,j,p\} \in E$ refers to an edge between nodes i and j, with p being an index that distinguishes parallel edges. For each edge $\{i,j,p\} \in E$, the coefficient f_{ijp} can be interpreted as its installation cost, and c_{ijp} as the cost of routing flow on the edge once installed.

Proposition 1. $Z_{2.15}^* \le Z_{2.16}^* \le (1+\epsilon) Z_{2.15}^*$. *This ratio can be achieved by introducing* $1 + \lceil \log_{1+4\epsilon+4\epsilon^2} B \rceil$ *pieces for each edge cost function.*

Proof. Since the objective is concave, it is well-known that problem (2.15) has an optimal solution that is a vertex of the feasible polyhedron [e.g. Bau58]. In this solution the flow of each commodity occurs on a tree [e.g. Sch03]. Consequently, any nonzero flow on any edge will be at least 1. On the other hand, the flow on any edge will be at most B. The approximation result now follows from Theorem 1 and Lemma 11. □

The special structure of the problem allows us to increase the number of edges by a factor of only $1 + \lceil \log_{1+4\epsilon+4\epsilon^2} B \rceil$, which is much less than the factor obtained for general polyhedra.

2.4.1 Computational Results

We present computational results for uncapacitated multicommodity flow problems with complete uniform demand. We have generated the instances based on [BMW89] as follows. To ensure feasibility, for each problem we first generated a random spanning tree. Then we added the desired number of edges between nodes selected uniformly at random. For each number of nodes, we considered a dense network with $\frac{n^2}{4}$ edges, and a sparse network with $3n$ edges. For each network thus generated, we have considered two cost structures.

The first cost structure models moderate economies of scale. We assigned to each edge $\{i,j\} \in E$ a cost function of the form $a + b(x_{ij})^c$, with a, b, and c randomly generated from uniform distributions over $[0.1, 10]$, $[0.33, 33.4]$, and $[0.8, 0.99]$. For an average cost function from this family, the marginal cost decreases by approximately 30% as the flow on an edge increases from 25 to 1,000. The second cost structure models strong economies of scale. The cost functions are as in the first case, except that c is sampled from a uniform distribution over $[0.0099, 0.99]$. In this case, for an average cost function, the marginal cost decreases by approximately 84% as the flow on an edge increases from 25 to 1,000. Note that on an undirected network with n nodes, there is an optimal solution with the flow on an edge in $\{0, 1, \ldots, n(n-1)\}$.

Table 2.1 specifies the problem sizes. Note that although the individual dimensions of the problems are moderate, the resulting number of variables is large, since a problem with n nodes and m edges yields $n(n-1)m$ flow variables. The largest problems we solved have 80 nodes, 1,580 edges, and 6,320 commodities. To approach them with an MIP solver, these problems would require 1,580 binary variables, 9,985,600 continuous variables and 10,491,200 constraints, even if we replaced the concave functions by fixed charge costs.

We chose $\epsilon = 0.01 = 1\%$ for the piecewise linear approximation. After applying our piecewise linear approximation technique, we have reduced the total number of pieces further by noting that close to 1, our approach introduced tangents on a grid denser than the uniform grid $1, 2, 3, \ldots$ For each problem, we have reduced the number of pieces per cost function by approximately 47 by using the uniform grid close to 1, and the grid generated by our approach elsewhere.

We used an improved version of the dual ascent method described by Balakrishnan et al. [BMW89] (also known as the primal-dual method [GW97]) to solve the resulting problems. The method produces a feasible solution, whose cost we denote by $Z^{\mathrm{DA}}_{2.16}$, to problem (2.16) and a lower bound $Z^{\mathrm{LB}}_{2.16}$ on the optimal value of problem (2.16). The lower bound allows us to obtain an instance-dependent error bound $\epsilon_{\mathrm{DA}} := \frac{Z^{\mathrm{DA}}_{2.16}}{Z^{\mathrm{LB}}_{2\ 16}} - 1$ for this solution, with respect to the piecewise linear problem. We can then obtain an instance-dependent error bound $\epsilon_{\mathrm{ALL}} := (1+\epsilon)(1+\epsilon_{\mathrm{DA}}) - 1$ for this solution with respect to the original problem.

Table 2.2 summarizes the computational results. We performed all the computations on a Pentium Xeon 2.8 GHz. For each problem size and cost structure, we have averaged the error bound, computational time, and number of edges in the computed solution over 3 randomly-generated instances.

We obtained average error bounds of 3.62% for problems with moderate economies of scale, and 4.20% for problems with strong economies of scale. This difference in average error bound is consistent with previous reports in the literature for fixed-charge functions, in which problems with higher fixed to variable cost ratios, and thus stronger economies of scale, have been found harder to solve [BMW89, HS89]. Note that the solutions to problems with moderate economies of scale have more edges than those to problems with strong economies of scale; in fact, in the latter case, the edges always form a tree.

To the best of our knowledge, the literature does not contain exact or approximate computational results for concave cost multicommodity network flow problems of this size.

#	n	m	K	Flow Variables	Pieces
1	10	30	90	8,100	41
2	20	60	380	22,800	77
3	20	95	380	36,100	77
4	30	90	870	78,300	98
5	30	215	870	187,050	98
6	40	120	1,560	187,200	113
7	40	390	1,560	608,400	113
8	50	150	2,450	367,500	124
9	50	610	2,450	1,494,500	124
10	60	180	3,540	637,200	133
11	60	885	3,540	3,132,900	133
12	70	210	4,830	1,014,300	141
13	70	1,205	4,830	5,820,150	141
14	80	240	6,320	1,516,800	148
15	80	1,580	6,320	9,985,600	148

Table 2.1: Network sizes. The column "Pieces" indicates the number of pieces in each piecewise linear function resulting from the approximation.

Bell and Lamar [BL97] propose an exact branch-and-bound approach for *single-commodity* flows, and present computational results on networks with at most 20 nodes and 96 arcs. Fontes et al. [FHC03] propose a heuristic approach for single-commodity flows, and present computational results on networks with up to 50 nodes and 200 edges. For a more detailed survey of previous work, see Section 1.2.

2.5 Facility Location

Next, we illustrate our approach on the concave cost facility location problem, defined in Section 1.1. Recall that the problem can be formulated as follows:

$$\min \sum_{j=1}^{n} \phi_j \left(\sum_{i=1}^{m} d_i x_{ij} \right) + \sum_{j=1}^{n} \sum_{i=1}^{m} c_{ij} d_i x_{ij}, \tag{2.17a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \qquad i \in [n], \tag{2.17b}$$

$$x_{ij} \geq 0, \qquad i \in [m], j \in [n]. \tag{2.17c}$$

As before, we assume that $\phi_j : \mathbb{R}_+ \to \mathbb{R}_+$ are nondecreasing concave functions, and that the connection costs $c_{ij} \geq 0$ obey the metric inequality. The intuition behind variables and coefficients is the same as in Section 1.1.

#	Moderate economies of scale				Strong economies of scale			
	Time	Sol. Edges	$\epsilon_{DA}\%$	$\epsilon_{ALL}\%$	Time	Sol. Edges	$\epsilon_{DA}\%$	$\epsilon_{ALL}\%$
1	0.26s	14	0.41	1.41	0.4s	9	0.35	1.35
2	7.57s	31	1.45	2.46	10.6s	19	1.06	2.07
3	8.77s	25.3	1.20	2.21	18.3s	19	3.38	4.42
4	48s	44	1.95	2.96	43.1s	29	1.18	2.20
5	1m33s	43.6	2.16	3.19	1m40s	29	3.50	4.54
6	3m29s	61.6	2.47	3.49	1m46s	39	2.20	3.22
7	6m49s	59	3.24	4.28	4m23s	39	3.17	4.21
8	9m16s	79	2.22	3.24	4m20s	49	3.42	4.46
9	20m51s	74.6	3.10	4.13	8m35s	49	4.22	5.26
10	21m10s	95	2.58	3.61	6m42s	59	3.27	4.30
11	56m58s	95.6	3.64	4.68	16m31s	59	4.25	5.29
12	40m42s	101.6	2.85	3.87	8m32s	69	3.77	4.81
13	1h47m	115.6	4.19	5.24	25m34s	69	4.98	6.03
14	1h18m	127.6	2.82	3.84	13m43s	79	4.10	5.14
15	3h3m	129.3	4.59	5.64	36m2s	79	4.68	5.73
Average			2.59	**3.62**			3.17	**4.20**

Table 2.2: Computational results. The values in column "Sol. Edges" represent the number of edges with positive flow in the obtained solutions.

Again we can assume without loss of generality that all coefficients d_i and c_{ij} are integral. Let $D = \sum_{i=1}^{n} d_i$ be the total demand.

As shown in Section 1.1, if we approximate ϕ_i by piecewise linear concave functions with Q pieces, then the resulting problem can be reduced to the classical facility location problem with Qn facilities:

$$\min \sum_{j=1}^{n}\sum_{p=1}^{P} f_j^p y_j^p + \sum_{j=1}^{n}\sum_{p=1}^{P}\sum_{i=1}^{m}(s_j^p + c_{ij})d_i x_{ij}^p, \tag{2.18a}$$

$$\text{s.t.} \sum_{j=1}^{n}\sum_{p=1}^{P} x_{ij}^p = 1, \qquad i \in [m], \tag{2.18b}$$

$$0 \le x_{ij}^p \le y_j^p, \qquad i \in [m], j \in [n], p \in \{0,\dots,P\}, \tag{2.18c}$$

$$y_j^p \in \{0,1\}, \qquad j \in [n], p \in \{0,\dots,P\}. \tag{2.18d}$$

Proposition 2. *$Z^*_{2.17} \le Z^*_{2.18} \le (1+\epsilon)Z^*_{2.17}$. This ratio can be achieved by introducing $1 + \lceil \log_{1+4\epsilon+4\epsilon^2} D \rceil$ pieces for each facility cost function.*

Proof. Since the objective is concave, problem (2.17) has an optimal solution that is a vertex of the feasible polyhedron [e.g. Bau58]. In this solution, the utilization of each

concave resource is at least 1. At the same time, the utilization of any resource does not exceed D. Therefore, the result follows from Theorem 1 and Lemma 11. □

To conclude our derivation, consider the 1.4991 approximation algorithm of Byrka for the classical facility location problem [Byr07]. By applying our technique together with this algorithm, we obtain the following result.

Theorem 3. *There exists a* $1.4991+\epsilon$ *approximation algorithm for the concave cost facility location problem. In particular, by taking* $\epsilon = 0.0009$*, we can obtain a 1.5 approximation algorithm.*

We can similarly apply our technique together with other approximation algorithms for facility location. For example, by applying it with the 1.52 approximation algorithm of Mahdian et al [MYZ06] and the 1.61 approximation algorithm of Jain et al [JMM+03], we obtain $1.52+\epsilon$ and $1.61+\epsilon$ approximation algorithm for concave cost facility lcoation.

Chapter 3

Primal-Dual Algorithms

In Chapter 2, we developed a general technique for approximating optimization problems with separable concave objectives by optimization problems with piecewise-linear objectives. Our key result is obtained when we are minimizing a nonnegative nondecreasing cost function over a polyhedron in $\mathbb{R}^n_+$, and would like the resulting problem to provide a $1+\epsilon$ approximation to the original problem in optimal cost. In this case, the size of the resulting problem is polynomial in the size of the original problem and linear in $1/\epsilon$. This bound implies that a variety of polynomial-time heuristics, approximation algorithms, and exact algorithms for fixed-charge problems immediately yield polynomial-time heuristics, approximation algorithms, and fully polynomial-time approximation schemes for the corresponding concave cost problems.

However, the piecewise-linear approach developed in Chapter 2 cannot fully address several questions involving the minimization of a separable concave objective over a polyhedron. First, the approach adds a relative error of $1+\epsilon$ in optimal cost, yet for some problems we would like to compute an exact optimum. For example, using our approach together with an exact algorithm for the classical lot-sizing problem, we can obtain a fully polynomial-time approximation scheme for the concave cost lot-sizing problem. However, there are exact algorithms for concave cost lot-sizing, both for the setting without backlogging [e.g. WW58, Wag60], and with backlogging [e.g. Zan66a, AP93]. Hence, fully polynomial-time approximation schemes for concave cost lot-sizing are of limited interest, and instead it is of interest to obtain improved exact algorithms.

Second, for some problems, after we approximate the concave cost problem by a piecewise linear problem, the resulting problem does not reduce in polynomial size to the corresponding fixed-charge problem. As a result, we cannot obtain polynomial-time heuristics, approximation algorithms, and fully polynomial-time approximation schemes for the concave cost problem through the technique of Chapter 2. An example is the concave cost joint replenishment problem (JRP). There are two obstacles to reducing the piecewise-linear concave cost JRP to the classical JRP. First, even if only one individual ordering cost function is piecewise-linear concave, the resulting problem need not satisfy the classical JRP's cost assumptions. Second, even if one sets aside the cost assumptions, a JRP with piecewise-

linear individual ordering costs can be reduced only to an exponentially-sized classical JRP. These obstacles are described in more detail in Section 1.4.

Third, suppose that we seek to compute near-optimal solutions to a concave cost problem by performing a $1 + \epsilon$ piecewise-linear approximation and then using a near-optimal computational procedure for the resulting fixed-charge problem. Since we seek near-optimal solutions, we are facing a tradeoff between selecting a larger value of ϵ and introducing a significant additional approximation error, or selecting a smaller value of ϵ and having to solve significantly larger problems produced by the approximation. For example in Section 2.4, the computational procedure used to solve the problems resulting from the approximation yielded an average approximation guarantee of 3.17%. However, since we chose $\epsilon = 1\%$, the resulting overall approximation guarantee averaged 4.20%. On the other hand, as can be seen from Table 2.2, choosing a significantly smaller value of ϵ would impact our ability to solve the largest instances.

The difficulties raised by these three questions are inherent in any piecewise-linear approximation approach, and cannot be addressed fully by devising improved piecewise-linear approximation methods. Therefore, we next turn to algorithms that operate directly on problems with concave cost functions.

3.1 Literature Review

There is a wide variety of algorithms that operate directly on problems with concave cost functions. In this section, we provide a brief survey of representative research directions.

A significant body of literature on concave cost problems is dedicated to algorithms that solve the problem exactly or to within a specified approximation error, employing branch-and-bound, cutting planes, as well as other methods. Pardalos and Rosen [PR86] provide a survey of earlier work in this direction. A survey on the application of these methods to the concave cost multicommodity flow problem can be found in Section 1.2. In brief, from a theoretical point of view, the surveyed algorithms do not have polynomial bounds on their running times. In computational experiments, these algorithms are successful on small to medium scale problems, however, once the problem sizes increase further, the reported computational times suggest an exponential dependendence of running time on problem size.

Another approach employed in the literature is heuristics. A survey of heuristics for the concave cost multicommodity flow problem is also presented in Section 1.2. To summarize, several authors obtain optimal or near-optimal results for small to medium scale problems. However, as the problem size increases, the results are inconclusive.

Mahdian and Pal [MP03] employed local search to obtain a theoretical result. More specifically, they obtained a $3 + \epsilon$ approximation algorithm for the concave cost facility location problem. Their analysis is based in part on the analysis of the local search algorithm by Charikar and Guha [CG99, CG05].

When the concave cost problems have unit demands, a wider variety of results become available. Hajiaghayi et al. [HMM03] obtained a 1.861 approximation algorithm for the

concave cost facility location problem. A number of results become available due to the fact that, as noted in Chapter 1, many concave cost problems with unit demands can be reduced to the corresponding classical problems. For example, Hajiaghayi et al [HMM03] and Mahdian et al. [MYZ06] described how to obtain a 1.52 approximation algorithm for concave cost facility location with unit demands using this approach.

Except for problems with unit demand, previously there have not been techniques for concave cost problems that would parallel the primal-dual framework for classical combinatorial optimization problems. Yet primal-dual algorithms account for important contributions to classical combinatorial optimization problems, including exact algorithms [e.g. LRS06], constant factor approximation algorithms [e.g. JMM+03, LRS06], and efficient heuristics [e.g. BMW89].

After completion of this research, we learned of the independent research of Romeijn et al. [RSSZ07]. They develop 1.61 and 1.52 approximation algorithms for the concave cost facility location problem by considering the corresponding algorithms for the classical facility location problem [JMM+03, MYZ06] through a greedy perspective. Since in this chapter we will consider these algorithms through a primal-dual perspective, establishing a connection between the research of Romeijn et al. and ours is in turn an interesting question.

3.1.1 Our Contribution

In this chapter we present an algorithm design technique for a class of concave cost problems. The technique yields a strongly polynomial primal-dual algorithm for a concave cost problem whenever such an algorithm exists for the corresponding fixed-charge problem. The resulting algorithm runs directly on concave problems, and thus avoids the approximation error/running time tradeoff of piecewise linear approximation. On the other hand, the resulting algorithm can be viewed as the original algorithm running on an exponentially or infinitely-sized fixed-charge problem. Therefore, constant factor approximation algorithms for classical problems yield constant factor approximation algorithms for the corresponding concave cost problems, with the approximation ratio preserved. Certain non-constant approximation ratios are also preserved, and exact algorithms yield exact algorithms. Since the execution of the new algorithm mirrors the execution of the original algorithm on an exponentially or infinitely-sized classical problem, we can also expect empirical properties, such as the near-optimal computational performance of heuristics, to be preserved in many cases.

We develop our technique on the basis of the concave cost facility location problem. In Section 3.2 we describe preliminary concepts. In Section 3.3 we obtain the key technical insights upon which our technique is based. Also in this section, we obtain a 1.61 approximation algorithm for the concave cost facility location problem, with a running time of $O(m^3n + mn \log n)$, and a 1.861 approximation algorithm with a running time of $O(m^2n + mn \log n)$. Here m denotes the number of customers and n the number of facilities. Previously, the best approximation ratio for this problem was $3 + \epsilon$, provided by the local search algorithm of Mahdian and Pal [MP03].

The technique introduced in Chapter 2 can provide an $1.4991 + \epsilon$ approximation algorithm for this problem, as well a variety of algorithms with higher approximation ratios. However, since these algorithms rely on piecewise-linear approximations, their running times depend on $1/\epsilon$. Their running times also depend on $\log \sum_{i=1}^{m} d_i$, where d_i are the customer demands. The primal-dual algorithms developed in this chapter have the advantage of strongly polynomial running times, with no dependence on an ϵ parameter. The work of Romeijn et al. [RSSZ07] provides 1.61 and 1.52 approximation algorithms with strongly polynomial running times of $O(m^3 n \log n)$ and $O(m^2 n \max\{m, n\} \log n)$. A more detailed literature review of approximation algorithms for concave cost facility location can be found in Section 1.1.

In Section 3.4 we obtain a new exact algorithm for the lot-sizing problem with concave ordering costs. The running time of the algorithm on a problem with n time periods is $O(n^2)$, which matches that of the fastest previous algorithm [e.g. AP93]. Here our goal is not to improve on previous exact algorithms for concave cost lot-sizing. Rather, we will use this algorithm to simplify the exposition of a primal-dual approximation algorithm for the concave cost JRP in the next section.

In Section 3.5, we obtain a 4 approximation algorithm for the joint replenishment problem (JRP) with concave individual ordering costs. This is the first constant factor approximation algorithm for the JRP with concave individual ordering costs. Previously, algorithms with constant worst-case performance guarantees for the JRP with concave costs were limited to more restricted models. Section 1.4 contains a more detailed literature review of approximation algorithms for the concave cost JRP.

3.2 Preliminaries

We first develop our technique on the basis of the facility location problem, and then apply it to other problems. As before, let $[n] = \{1, \ldots, n\}$. Recall from Section 1.1 that the concave cost facility location problem can be formulated as follows:

$$\min \sum_{j=1}^{n} \phi_j \left(\sum_{i=1}^{m} d_i x_{ij} \right) + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} d_i x_{ij}, \tag{3.1a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \qquad i \in [m], \tag{3.1b}$$

$$x_{ij} \geq 0, \qquad i \in [m], j \in [n]. \tag{3.1c}$$

The meaning of coefficients and variables is the same as in Section 1.1. As before, we assume that the cost functions $\phi_j : \mathbb{R}_+ \to \mathbb{R}_+$ are concave and nondecreasing, and that the connection costs c_{ij} are nonnegative and obey the metric inequality.

Without loss of generality we assume that all coefficients d_i and c_{ij} are integral. Let $D = \sum_{i=1}^{n} d_i$ be the total demand.

The classical facility location problem, also defined in Section 1.1, is a special case of the concave cost facility location problem and can be formulated as a mixed-integer program as follows:

$$\min \sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij} d_i x_{ij}, \tag{3.2a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \qquad i \in [m], \tag{3.2b}$$

$$0 \le x_{ij} \le y_j, \qquad i \in [m], j \in [n], \tag{3.2c}$$

$$y_j \in \{0,1\}, \qquad i \in [m], j \in [n]. \tag{3.2d}$$

We will make use of the well-known reduction from piecewise-linear concave cost facility location to classical facility location. Let the concave functions ϕ_j be piecewise linear on $(0, +\infty)$ with P pieces. Then the functions can be written as

$$\phi_j(\xi_j) = \begin{cases} \min\{f_{jp} + s_{jp}\xi_j : p \in [P]\}, & \xi_j > 0, \\ 0, & \xi_j = 0. \end{cases} \tag{3.3}$$

And the facility location problem with these cost functions can be reformulated as:

$$\min \sum_{j=1}^{n}\sum_{p=1}^{P} f_{jp} y_{jp} + \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{p=1}^{P} (c_{ij} + s_{jp}) d_i x_{ijp}, \tag{3.4a}$$

$$\text{s.t.} \sum_{j=1}^{n}\sum_{p=1}^{P} x_{ijp} = 1, \qquad i \in [m], \tag{3.4b}$$

$$0 \le x_{ijp} \le y_{jp}, \qquad i \in [m], j \in [n], p \in [P], \tag{3.4c}$$

$$y_{jp} \in \{0,1\}, \qquad i \in [m], j \in [n], p \in [P]. \tag{3.4d}$$

This mixed-integer program is a classical facility location problem with Pn facilities and m customers. Every piece p in the cost function ϕ_j of every facility j in the original concave cost problem corresponds to a facility $\{j, p\}$ in the new problem. The new facility has opening cost f_{jp}. The customer set is the same, and the connection cost from facility $\{j, p\}$ to customer i is $c_{ij} + s_{jp}$. We describe the reduction more formally in Section 1.1.

3.2.1 A Primal-Dual Algorithm

In this section, we briefly describe the 1.61 approximation algorithm for classical facility location due to Jain et al. [JMM+03]. In the following sections, we will obtain a 1.61 approximation algorithm for concave cost facility location based on this algorithm. We assume the reader is familiar with the primal-dual method for approximation algorithms [see e.g. GW97].

Consider the LP relaxation of problem (3.2) obtained by replacing the constraints $y_j \in \{0,1\}$ with $y_j \geq 0$. The dual of this LP relaxation is:

$$\max \sum_{i=1}^{n} v_i, \tag{3.5a}$$

$$\text{s.t. } v_i \leq c_{ij} d_i + w_{ij}, \qquad i \in [n], j \in [n], \tag{3.5b}$$

$$\sum_{i=1}^{m} w_{ij} \leq f_j, \qquad j \in [n], \tag{3.5c}$$

$$w_{ij} \geq 0, \qquad i \in [m], j \in [n]. \tag{3.5d}$$

Since w_{ij} are not present in the objective, we can assume that they are as small as is possible without violating constraint (3.5b). In other words, we assume the invariant $w_{ij} = \max\{0, v_i - c_{ij} d_i\}$. We will refer to dual variable v_i as the *budget* of customer i. If $v_i \geq c_{ij} d_i$, we say that customer i *contributes* to the fixed cost f_j of facility j, and w_{ij} is its contribution. A facility j is *tight* if its fixed charge is fully covered by customer contributions, i.e. $\sum_{i=1}^{n} w_{ij} = f_j$. A facility is *over-tight* if $\sum_{i=1}^{n} w_{ij} > f_j$.

Let (x, y) be an integral primal feasible solution, and (v, w) be a dual feasible solution. The primal complementary slackness constraints are:

$$x_{ij}(v_i - c_{ij} d_i - w_{ij}) = 0, \qquad i \in [m], j \in [n], \tag{3.6a}$$

$$y_j \left(\sum_{i=1}^{n} w_{ij} - f_j \right) = 0, \qquad j \in [n]. \tag{3.6b}$$

Constraint (3.6a) says that customer i can connect to facility j (i.e. $x_{ij} = 1$) in the primal solution only if j is the closest to i with respect to the modified connection costs $c_{ij} + w_{ij}/d_i$. Constraint (3.6b) says that facility j can be opened in the primal solution (i.e. $y_j = 1$) only if it is tight in the dual solution.

The algorithm of Jain et al. starts with dual feasible solution $(v, w) = 0$ and iteratively updates it, while maintaining dual feasibility and increasing the dual objective. (The increase in the objective is not necessarily monotonic.) At the same time, guided by the complementary slackness constraints, the algorithm constructs an integral primal solution. The algorithm concludes when the integral primal solution becomes feasible; at this point the dual feasible solution serves as a lower bound on the optimal value.

We introduce the notion of time, and associate to each step of the algorithm the time when it ocurred. Time will be denoted by t.

ALGORITHM FLPD($m, n \in \mathbb{Z}_+$; $c \in \mathbb{R}_+^{mn}$, $f \in \mathbb{R}_+^n$, $d \in \mathbb{R}_+^m$)

(1) Start at time $t = 0$ and with the dual solution $(v, w) = 0$. All facilities are closed and all customers are unconnected, i.e. $(x, y) = 0$.

(2) **While** there are unconnected customers:

(3) Increase t continuously. At the same time increase v_i and w_{ij} for unconnected customers i so as to maintain $v_i = d_i t$ and $w_{ij} = \max\{0, v_i - c_{ij} d_i\}$. The increase stops when a facility becomes tight, or a customer begins contributing to a previously open facility.

(4) If a facility j became tight, open it. For each customer i contributing to j, connect i to j and set $v_i = c_{ij} d_i$. Then set $w_{ij'} = \max\{0, v_i - c_{ij'} d_i\}$ for all closed facilities j'. If i was previously connected to a different facility j'', disconnect i from j''.

(5) If a customer i began contributing to a previously open facility j, connect i to j and set $v_i = c_{ij} d_i$. Then set $w_{ij'} = \max\{0, v_i - c_{ij'} d_i\}$ for all closed facilities j'.

(6) For each customer i, set $v_i = c_{ij} d_i + w_{ij}$, where j is the facility that customer i is connected to. Return (x, y) and (v, w).

In case of a tie between tight facilities in step (4), between customers in step (5), or between steps (4) and (5), we break the tie arbitrarily. Depending on the customers that remain unconnected, in the next iteration of loop (2), another one of the facilities involved in the tie may open immediately, or another one of the customers involved in the tie may connect immediately.

Theorem 4 (JMM+03). *Algorithm FLPD is a 1.61 approximation algorithm for the classical facility location problem.*

Note that we use a different MIP formulation in our presentation from that in [JMM+03]. However both the formulation and the algorithm are equivalent to those in the original presentation.

3.3 The Technique

Consider the concave cost facility location problem with m customers and n facilities. Assume that the functions ϕ_j are given by an oracle that returns $\phi_j(\xi_j)$ and $\phi_j'(\xi_j)$, in time $O(1)$. If we would like to avoid using derivatives, for example when it is computationally expensive, we can use the quantity $\phi_j(\xi_j + 1) - \phi_j(\xi_j)$ instead, as all demands are integer.

We interpret each concave function ϕ_j as a piecewise-linear function with an infinite number of pieces. We need to consider the function only up to the total demand D. Formally, for each $p \in (0, D]$, we introduce a tangent $f_{jp} + s_{jp}\xi_j$ to ϕ_j at p. When the derivative $\phi_j'(p)$ exists, the coefficients f_{jp} and s_{jp} are given by

$$s_{jp} = \phi_j'(p), \qquad f_{jp} = \phi_j(p) - p s_{jp}. \tag{3.7}$$

If the derivative does not exist, we take the derivative from the right of ϕ_j at p, that is

$s_{jp} = \lim_{\xi_j \to p+} \frac{\phi_j(p) - \phi_j(\xi_j)}{p - \xi_j}$. The derivative from the right always exists at $p > 0$, since ϕ_j is concave on $[0, +\infty)$.

To simplify the notation in what follows, we must consider a technical detail. Note that $\lim_{q \to 0+} s_{jq}$ either exists or is $+\infty$ because s_{jp} are nonincreasing in p, and $\lim_{q \to 0+} f_{jq}$ exists because f_{jp} are nondecreasing in p and bounded from below. We assume that $\lim_{q \to 0+} s_{jq}$ exists, and introduce a tangent at $p = 0$ that is the limit of the tangents at q as $q \to 0$. More precisely, we take $s_{j0} = \lim_{q \to 0+} s_{jq}$ and $f_{j0} = \lim_{q \to 0+} f_{jq}$. Our technique also applies when the $\lim_{q \to 0+} s_{jq}$ is $+\infty$, in which case we introduce tangents only for $p > 0$ and proceed in similar fashion.

The functions ϕ_j can now be expressed as:

$$\phi_j(\xi_j) = \begin{cases} \min\{f_{jp} + s_{jp}\xi_j : p \in [0, D]\}, & \xi_j > 0, \\ 0, & \xi_j = 0. \end{cases} \tag{3.8}$$

When ϕ_j is linear on a segment $[\xi_1, \xi_2]$, we obtain the same tangent at any point $p \in [\xi_1, \xi_2)$, that is $f_{jp} = f_{jq}$ and $s_{jp} = s_{jq}$ for any $p, q \in [\xi_1, \xi_2)$. For ease of notation, we will consider f_{jp} and s_{jp} for all $p \geq 0$, regardless of the shape of ϕ_j.

We apply Lemma 1, and obtain a classical facility location problem with m customers and an infinite number of facilities. Each tangent p to cost function $\phi_j(\xi_j)$ of facility j in the concave cost problem corresponds to a facility $\{j, p\}$ in the resulting classical facility location problem. Due to their origin, we will sometimes call the facilities in the resulting problem tangents.

The infinitely-sized problem is:

$$\min \sum_{j=1}^{n} \sum_{p \in [0,D]} f_{jp} y_{jp} + \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{p \in [0,D]} (c_{ij} + s_{jp}) d_i x_{ijp}, \tag{3.9a}$$

$$\text{s.t.} \sum_{j=1}^{n} \sum_{p \in [0,D]} x_{ijp} = 1, \qquad i \in [m], \tag{3.9b}$$

$$0 \leq x_{ijp} \leq y_{jp}, \qquad i \in [m], j \in [n], p \in [0, D], \tag{3.9c}$$

$$y_{jp} \in \{0, 1\}, \qquad i \in [m], j \in [n], p \in [0, D]. \tag{3.9d}$$

Of course, we cannot run Algorithm FLPD on this problem directly, as it is infinitely-sized. Instead, we will show how to execute Algorithm FLPD on the infinitely-sized instance implicitly. Formally, we will devise an algorithm that produces the same assignment of customers to facilities as if Algorithm FLPD were run on the infinitely-sized problem, in time polynomial in the input size of the original problem. Thereby, we will obtain a 1.61 approximation algorithm for the original problem.

The LP relaxation is obtained by relaxing the integrality constraints (3.9d) to say $y_{jp} \geq 0$. The dual of the LP relaxation is:

$$\max \sum_{i=1}^{n} v_i, \tag{3.10a}$$

$$\text{s.t. } v_i \leq (c_{ij} + s_{jp})d_i + w_{ijp}, \qquad i \in [n], j \in [n], p \in [0, D], \tag{3.10b}$$

$$\sum_{i=1}^{m} w_{ijp} \leq f_{jp}, \qquad j \in [n], p \in [0, D], \tag{3.10c}$$

$$w_{ijp} \geq 0, \qquad i \in [m], j \in [n], p \in [0, D]. \tag{3.10d}$$

Since the LP relaxation and its dual are infinitely-sized, the strong duality property between the LP relaxation and its dual does not hold automatically, as in the finite LP case. However, we do not need strong duality for our approach. We only rely on the optimal value of MIP (3.9) being greater than or equal to that of the LP relaxation, and on weak duality between the LP relaxation and its dual (3.10).

3.3.1 Analysis of a Single Facility

In this section, we consider the special case when the original concave cost problem has only one facility. To simplify the notation, we omit the facility subscript j.

Imagine that we are at the beginning of step (3) of the algorithm. To execute this step, we need to compute the time t^* when the increase in the dual variables stops. If the increase stops due to a new tangent p^* becoming tight, we need to compute p^*. If the increase stops because a customer begins to contribute to a previously open tangent, we need to compute the customer and tangent.

We assume that a new tangent will become tight before an unconnected customer will begin contributing to a previously open tangent. In other words, the increase always stops due to a new tangent p^* becoming tight. Also assume that $c_i = 0$ for $i \in [m]$. We will remove both of these assumptions in Section 3.3.3.

Without loss of generality, we assume $t = 0$ at the beginning of the step. We imagine that t is increasing to $+\infty$, and seek to compute the first tangent to become tight. The customer budgets start at v_i and increase with time at rates δ_i. In other words, at time t, the budget for customer i has increased to $v_i + \delta_i t$. Connected customers can be modeled by taking $\delta_i = 0$, and unconnected customers by taking $\delta_i = d_i$. Denote the set of unconnected customers by U and the set of connected customers by C, and let $\mu = |U|$.

First, consider the case when all customers have zero starting budgets.

Lemma 12. *Let all customers have zero starting budgets and the budgets v_i for unconnected customers increase at rates d_i. The tangent that becomes tight first is unique and is given by $p^* = \sum_{i \in U} d_i$, while the time when this occurs is given by $t^* = s_{p^*} + \frac{f_{p^*}}{p^*}$.*

Proof. For a given tangent p, the time when it becomes tight is $s_p + \frac{f_p}{\sum_{i\in U} d_i}$. Therefore,

$$p^* = \operatorname*{argmin}_{p\geq 0} \left\{ s_p + \frac{f_p}{\sum_{i\in U} d_i} \right\} = \operatorname*{argmin}_{p\geq 0} \left\{ s_p \sum_{i\in U} d_i + f_p \right\}. \tag{3.11}$$

The quantity $s_p \sum_{i\in U} d_i + f_p$ can be viewed as the value of the affine function $f_p + s_p\xi$ at $\xi = \sum_{i\in U} d_i$. Since $f_p + s_p\xi$ is tangent to $\phi(\xi)$, and $\phi(\xi)$ is concave,

$$f_p + s_p \sum_{i\in U} d_i \geq \phi\left(\sum_{i\in U} d_i\right) \quad \text{for} \quad p \geq 0. \tag{3.12}$$

On the other hand, for the tangent at $q := \sum_{i\in U} d_i$, we have $f_q + s_q \sum_{i\in U} d_i = \phi\left(\sum_{i\in U} d_i\right)$. Therefore, the tangent at q becomes tight first, at time $s_q + \frac{f_q}{q}$. (See Figure 3-1.)

Recall from the discussion after equation (3.8) that when $\phi(\xi)$ is linear on a segment, two different tangency points p and q may yield the same tangent $(f_p, s_p) = (f_q, s_q)$. Now consider a tangent p that is different from p^*, i.e. that has $(f_p, s_p) \neq (f_{p^*}, s_{p^*})$. For any such tangent p, the strict inequality $f_p + s_p \sum_{i\in U} d_i > \phi\left(\sum_{i\in U} d_i\right)$ holds. Therefore, the tangent that becomes tight first is unique. □

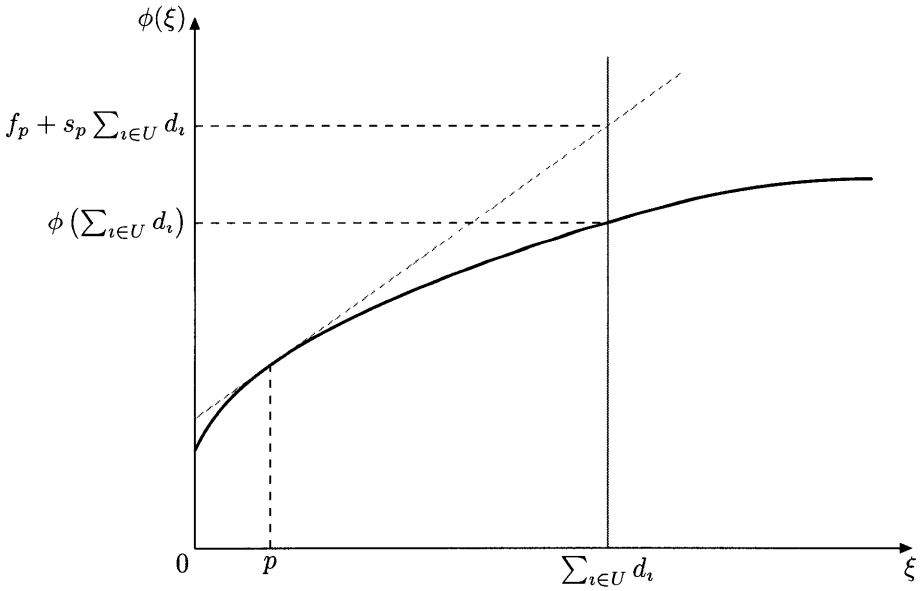

Figure 3-1: Illustration of the proof of Lemma 12.

Next, consider the case when customers have nonnegative starting budgets. Define

$$p_i(t) = \min\{p \in [0, D] : v_i + \delta_i t \geq s_p d_i\}, \qquad i \in [m], \tag{3.13}$$

If $v_i + \delta_i t < s_p d_i$ for every $p \in [0, D]$, let $p_i(t) = D$. Otherwise, the minimum is well-defined, since s_p is right-continuous in p.

Intuitively, $p_i(t)$ is the leftmost tangent to which customer i is contributing at time t. Note that s_p is decreasing in p, since ϕ is a concave function. Therefore, customer i contributes to every tangent to the right of $p_i(t)$, and does not contribute to any tangent to the left of $p_i(t)$. For any two customers i and j,

$$(v_i + t\delta_i)/d_i > (v_j + t\delta_j)/d_j \quad \Rightarrow \quad p_i(t) \leq p_j(t), \tag{3.14a}$$
$$(v_i + t\delta_i)/d_i = (v_j + t\delta_j)/d_j \quad \Rightarrow \quad p_i(t) = p_j(t), \tag{3.14b}$$
$$(v_i + t\delta_i)/d_i < (v_j + t\delta_j)/d_j \quad \Rightarrow \quad p_i(t) \geq p_j(t). \tag{3.14c}$$

Lemma 13. *Let customers have starting budgets v_i, the budgets of connected customers be constant, and the budgets of unconnected customers increase at rates d_i. The tangent p^* that becomes tight first, and the time t^* when this occurs can be computed in $O(m^2)$. If there is a tie, it is between at most m tangents.*

Proof. Assume without loss of generality that the set of customers is ordered so that customers $1, \ldots, \mu$ are unconnected, customers $\mu + 1, \ldots, m$ are connected, and

$$v_1/d_1 \geq v_2/d_2 \geq \cdots \geq v_\mu/d_\mu, \tag{3.15a}$$
$$v_{\mu+1}/d_{\mu+1} \geq v_{\mu+2}/d_{\mu+2} \geq \cdots \geq v_m/d_m. \tag{3.15b}$$

Note that $(v_i + t\delta_i)/d_i = v_i/d_i$ for connected customers, and $(v_i + t\delta_i)/d_i = v_i/d_i + t$ for unconnected ones. By property (3.14), it follows that $p_1(t) \leq p_2(t) \leq \cdots \leq p_\mu(t)$ and $p_{\mu+1}(t) \leq p_{\mu+2}(t) \leq \cdots \leq p_m(t)$. As t increases, $p_i(t)$ for $i \in C$ are unchanged, while $p_i(t)$ for $i \in U$ decrease. (See Figure 3-2.)

Let

$$I_k^{\mathrm{u}}(t) = [p_k(t), p_{k+1}(t)], \qquad 1 \leq k < \mu, \tag{3.16a}$$
$$I_l^{\mathrm{c}}(t) = [p_l(t), p_{l+1}(t)], \qquad \mu + 1 \leq l < m, \tag{3.16b}$$

with $I_0^{\mathrm{u}}(t) = [0, p_1(t)]$ and $I_\mu^{\mathrm{u}}(t) = [p_\mu(t), D]$, as well as $I_\mu^{\mathrm{c}}(t) = [0, p_{\mu+1}(t)]$ and $I_m(t) = [p_m(t), D]$. Consider the intervals

$$I_{kl}(t) = I_k^{\mathrm{u}}(t) \cap I_l^{\mathrm{c}}(t), \qquad 0 \leq k \leq \mu \leq l \leq m. \tag{3.17}$$

At any given time t, some of these intervals may be empty. As time increases, intervals may vary in size, empty intervals may become non-empty, and non-empty intervals may become empty. Together the intervals cover $[0, D]$.

Let p^*_{kl} be the tangent that becomes tight first on $I_{kl}(t)$, and t^*_{kl} be the time when this occurs. If a tangent never becomes tight on a given interval $I_{kl}(t)$, define $t^*_{kl} = +\infty$. Obviously, $t^*_{0l} = +\infty$, since on $I_{0l}(t)$ the contributions do not increase with time. However, other intervals may also have $t^*_{kl} = +\infty$, say because the interval becomes empty for times $[\theta, +\infty)$ and no tangents become tight on it before time θ. Clearly, $t^* = \min\{t^*_{kl} : 1 \le k \le \mu \le l \le m\}$ and $p^* = p^*_{\text{argmin}\{t^*_{kl}:1\le k\le\mu\le l\le m\}}$.

Fix a time t, and an interval $I_{kl}(t)$ that is non-empty at time t and has $k \ge 1$. Let us evaluate the total contribution received by a tangent $p \in I_{kl}(t)$ at time t. By the definition of $I_{kl}(t)$, tangent p is receiving contributions from unconnected customers $1, \ldots, k$ and connected customers $\mu + 1, \ldots, l$. Let $S(k,l) = \{1, \ldots, k\} \cup \{\mu + 1, \ldots, l\}$. The total contribution is

$$\sum_{i=1}^{k}(v_i + d_i t - s_p d_i) + \sum_{i=\mu+1}^{l}(v_i - s_p d_i) = \sum_{i\in S(k,l)}(v_i - s_p d_i) + t\sum_{i=1}^{k} d_i$$
$$= \sum_{i\in S(k,l)}(v_i + \alpha_{kl} t d_i - s_p d_i) = (\beta_{kl} + \alpha_{kl} t - s_p)\sum_{i\in S(k,l)} d_i = (\tau_{kl} - s_p)\sum_{i\in S(k,l)} d_i, \quad (3.18)$$

where $\alpha_{kl} := \sum_{i=1}^{k} d_i / \sum_{i\in S(k,l)} d_i$, $\beta_{kl} := \sum_{i\in S(k,l)} v_i / \sum_{i\in S(k,l)} d_i$, and $\tau_{kl} := \beta_{kl} + \alpha_{kl} t$. Note that $\sum_{i\in S(k,l)} d_i > 0$ and $\alpha_{kl} > 0$. The last expression in (3.18) can be interpreted as the total contribution to tangent p at time τ_{kl} in an alternate setting, where all starting budgets v_i are zero, and each customer $i \in S(k,l)$ is increasing its budget at rate d_i. The contribution at time t in the original setting is the same as the contribution at time τ_{kl} in the alternative setting for all tangents $p \in I_{kl}(t)$.

Let us compare the contributions in the alternative and original settings at time t for a tangent $p \notin I_{kl}(t)$. If p does not receive a positive contribution in the alternative setting, then the contribution is the original setting is no less than in the alternative setting. If p receives a positive contribution in the alternative setting, let $p \in I_{rs}(t)$. The contribution in the alternative setting is $(\tau_{kl} - s_p)\sum_{i\in S(k,l)} d_i = \sum_{i=1}^{k}(v_i + d_i t - s_p d_i) + \sum_{i=\mu+1}^{l}(v_i - s_p d_i)$, while the contribution in the original setting is $\sum_{i=1}^{r}(v_i + d_i t - s_p d_i) + \sum_{i=\mu+1}^{s}(v_i - s_p d_i)$. We will prove the inequality

$$(\tau_{kl} - s_p)\sum_{i\in S(k,l)} d_i = \sum_{i=1}^{k}(v_i + d_i t - s_p d_i) + \sum_{i=\mu+1}^{l}(v_i - s_p d_i)$$
$$\le \sum_{i=1}^{r}(v_i + d_i t - s_p d_i) + \sum_{i=\mu+1}^{s}(v_i - s_p d_i). \quad (3.19)$$

The difference between the two contributions can be written as

$$\sum_{i=r+1}^{k}(v_i + d_i t - s_p d_i) - \sum_{i=k+1}^{r}(v_i + d_i t - s_p d_i) + \sum_{i=s+1}^{l}(v_i - s_p d_i) - \sum_{i=l+1}^{s}(v_i - s_p d_i). \quad (3.20)$$

We will now examine the summations in this expression one by one:

1. $\sum_{i=r+1}^{k}(v_i + d_i t - s_p d_i)$. This summation is nonempty when $r < k$. In this case, in the original setting, customers $r+1, \ldots, k$ do not contribute to tangents in $I_{rs}(t)$ at time t. Therefore, $v_i + d_i t - s_p d_i \leq 0$ for $i = r+1, \ldots, k$, and the summation is nonpositive.

2. $-\sum_{i=k+1}^{r}(v_i + d_i t - s_p d_i)$. This summation is nonemtpy when $r > k$. In this case, in the original setting, customers $k+1, \ldots, r$ do contribute to tangents in $I_{rs}(t)$ at time t. Therefore, $v_i + d_i t - s_p d_i \geq 0$ for $i = k+1, \ldots, r$, and the summation is nonpositive.

3. $\sum_{i=s+1}^{l}(v_i - s_p d_i)$. This summation is nonemtpy when $s < l$. In this case, in the original setting, customers $s+1, \ldots, l$ do not contribute to tangents in $I_{rs}(t)$ at time t. Therefore, $v_i - s_p d_i \leq 0$ for $i = s+1, \ldots, l$, and the summation is nonpositive.

4. $-\sum_{i=l+1}^{s}(v_i - s_p d_i)$. This summation is nonempty when $l < s$. In this case, in the original setting, customers $l+1, \ldots, s$ do contribute to tangents in $I_{rs}(t)$ at time t. Therefore, $v_i - s_p d_i \geq 0$, and the summation is nonpositive.

Therefore the contribution in the original setting is greater than or equal to the contribution in the alternative setting. (The inequality also holds in the strict sense, since $p \in I_{rs}(t)$ and $p \notin I_{kl}(t)$, and thus at least one of these summations is nonempty and strictly negative. However, we do not need the inequality in the strict sense for this proof.)

In the alternative setting, we can apply Lemma 12 to compute the first tangent to become tight, and the time when this occurs. Denote the computed tangent and time by p'_{kl} and τ'_{kl}, respectively. Then, $p'_{kl} = \sum_{i \in S(k,l)} d_i$ and $\tau'_{kl} = s_{p'_{kl}} + \frac{f_{p'_{kl}}}{p'_{kl}}$. Let $t'_{kl} := \frac{\tau'_{kl} - \beta_{kl}}{\alpha_{kl}}$ be the time in the original setting that corresponds to time τ'_{kl} in the alternative setting. We shall use p'_{kl} and t'_{kl} to compute p^* and t^*. We distinguish the following cases.

Case 1: $t'_{kl} < 0$. Let τ^0_{kl} be the time in the alternative setting that corresponds to time 0 in the original setting. Since $\tau^0_{kl} > \tau'_{kl}$, tangent p'_{kl} is over-tight in the alternative setting at time τ^0_{kl}. Since the contribution to p'_{kl} is no less at time 0 in the original setting than at time τ^0_{kl} in the alternative setting, p'_{kl} is over-tight in the original setting at time 0. This is a contradiction, and thus this case cannot occur.

Case 3: $t'_{kl} \geq 0$ and $p'_{kl} \in I_{kl}(t'_{kl})$. Since p'_{kl} is the first tangent to become tight in the alternative setting, it is the first to become tight on $I_{kl}(\cdot)$ in the original setting. In other words, $p^*_{kl} = p'_{kl}$ and $t^*_{kl} = t'_{kl}$.

Case 4: $t'_{kl} \geq 0$ and $p'_{kl} \notin I_{kl}(t'_{kl})$. There are no tight tangents on $I_{kl}(t'_{kl})$ at time τ'_{kl} in the alternative setting, and thus there are none on $I_{kl}(t'_{kl})$ at time t'_{kl} in the original setting. Moreover, there are no tight tangents anywhere in the alternative setting at times $[0, \tau'_{kl})$, and therefore there are no tight tangents on $I_{kl}(t)$ at any time $t \in [0, t'_{kl})$. Therefore, $t^*_{kl} > t'_{kl}$.

On the other hand, since $p'_{kl} \notin I_{kl}(t'_{kl})$ is tight in the alternative setting at time τ'_{kl}, it is tight or over-tight at time t'_{kl} in the original setting. Consequently, there is an interval $I_{rs}(\cdot)$ such that $t^*_{rs} \le t'_{kl}$. We will employ the inequality $t^*_{rs} \le t'_{kl} < t^*_{kl}$ later in the proof.

Let $I_{kl}(t^*)$ be the interval for which $p^* \in I_{kl}(t^*)$; trivially $p^*_{kl} = p^*$ and $t^*_{kl} = t^*$. If $p'_{kl} \notin I_{kl}(t'_{kl})$, then there is another interval $I_{rs}(t'_{kl})$ such that $p'_{kl} \in I_{rs}(t'_{kl})$. By Case 3, $t^*_{rs} \le t'_{kl} < t^*_{kl}$, which is a contradiction because $t^*_{kl} = t^* \le t^*_{rs}$. Hence $p'_{kl} \in I_{kl}(t'_{kl})$, and therefore $p'_{kl} = p^*_{kl} = p^*$ and $t'_{kl} = t^*_{kl} = t^*$.

On the other hand, consider any interval $I_{kl}(t)$. If $p'_{kl} \in I_{kl}(t'_{kl})$, then $t'_{kl} = t^*_{kl} \ge t^*$. If $p'_{kl} \notin I_{kl}(t'_{kl})$, let $I_{rs}(t'_{kl})$ be the interval with $p'_{kl} \in I_{rs}(t'_{kl})$. Then by Case 3, $t'_{kl} \ge t^*_{rs} \ge t^*$. Hence, although t'_{kl} does not necessarily equal t^*_{kl} for every k and l, we have $t^* = \min\{t'_{kl} : 1 \le k \le \mu \le l \le m\}$ and $p^* = p'_{\arg\min\{t'_{kl}:1\le k\le\mu\le m\}}$.

To evaluate the running time needed to compute p^* and t^*, note that sorting the v_i requires $O(m \log m)$ time. Once the v_i's are sorted, we can compute all t'_{kl} and p'_{kl} in $O(m^2)$ time via Lemma 12. Therefore the total running time is $O(m^2)$.

To evaluate the number of ties, observe that at time t^*, the points $p_i(t)$ divide $[0, D]$ into at most $m + 1$ intervals, and at most m intervals receive customer contributions. By Lemma 12, there is at most one tight tangent on each of the m intervals, and therefore there are at most m tied tangents. □

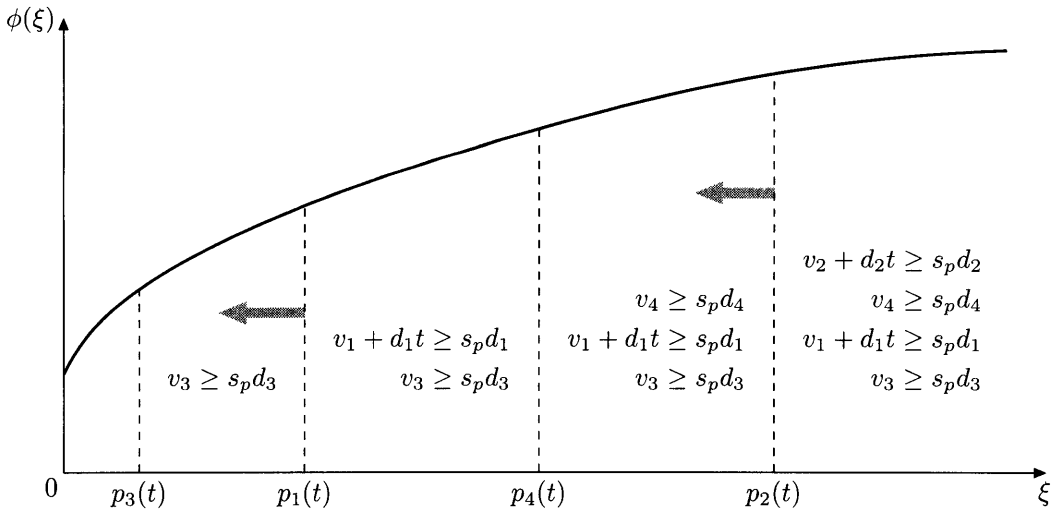

Figure 3-2: Illustration of the proof of Lemma 12. Here $U = \{1, 2\}$ and $C = \{3, 4\}$. The gray arrows show how $p_i(t)$ change as t increases.

Lemmas 12 and 13 contain the key insights that will enable us to obtain a variety of primal-dual algorithms for problems with concave costs. For some primal-dual algorithms

for specific problems, we will be able to perform the computation in Lemma 13 faster than $O(m^2)$, by taking advantage of the way each specific algorithm increases the dual variables.

3.3.2 Other Rules for Changing the Dual Variables

The following lemma is a generalization of Lemma 13. We consider the same setting, but in addition allow each customer to change its budget at an arbitrary rate δ_i. The rate need not equal d_i, and may even be negative.

We do not need this lemma to obtain any of the algorithms in this chapter. Rather, we present it here since it embodies a more general version of our approach, and may be useful in future primal-dual applications. As such, the reader may wish to skip this section.

Lemma 14. *Let customers have starting budgets v_i, and let the budgets change at rates δ_i. The tangent p^* that becomes tight first, and the time t^* when this occurs can be computed in $O(m^3)$. If there is a tie, it is between at most m tangents.*

Proof. For every time $t \geq 0$, we sort the quantities $p_1(t), p_2(t), \ldots, p_m(t)$ in nondecreasing order. Since δ_i need not equal d_i, this order may change as t increases from 0 to $+\infty$. By property (3.14), each time the order changes, at least one inequality of the form

$$(v_i + t\delta_i)/d_i \quad \text{vs.} \quad (v_j + t\delta_j)/d_j \tag{3.21}$$

changes its sign from $<$ to $>$ or vice-versa. Since each of the $m(m-1)/2$ inequalities changes its sign at most once, the order changes at most $m(m-1)/2$ times. Denote the times when the order changes by $\theta_1, \ldots, \theta_L$; also let $\theta_0 = 0$ and $\theta_{L+1} = +\infty$.

Fix a time $t \in [\theta_l, \theta_{l+1}]$, and assume w.l.o.g. that $p_1(t) \leq p_2(t) \leq \cdots \leq p_m(t)$. Also fix a tangent p and let k be such that $p \in [p_k(t), p_{k+1}(t))$. The total contribution received by tangent p at time t is

$$\sum_{i=1}^{k} (v_i + \delta_i t - s_p d_i) = (\beta_{kl} + \alpha_{kl} t - s_p) \sum_{i=1}^{k} d_i = (\tau_{kl} - s_p) \sum_{i=1}^{k} d_i, \tag{3.22}$$

where $\alpha_{kl} := \sum_{i=1}^{k} \delta_i / \sum_{i=1}^{k} d_i$, $\beta_{kl} := \sum_{i=1}^{k} v_i / \sum_{i=1}^{k} d_i$, and $\tau_{kl} := \beta_{kl} + \alpha_{kl} t$. We can discard the intervals with $\sum_{i=1}^{k} \delta_i \leq 0$, since a tangent will never become tight first in such an interval. Hence, we can assume that $\alpha_{kl} > 0$. As in Lemma 12, the last expression in (3.22) can be viewed as the total contribution to tangent p at time τ_{kl} in an alternative setting, where all starting budgets are zero, and customers $1, \ldots, k$ are increasing their budgets at rates $d_1, \ldots, d_k$. The contribution at time t in the original setting is the same as the contribution at time τ_{kl} in the alternative setting for all tangents in $[p_k(t), p_{k+1}(t)]$ whenever $t \in [\theta_l, \theta_{l+1}]$.

Let us compare the contributions in the alternative and original settings when $t \notin [\theta_l, \theta_{l+1}]$, or $p \notin [p_k(t), p_{k+1}(t)]$. If p does not receive a positive contribution in the alternative setting, the contribution in the original setting is no less. If p does receive a positive contribution in the alternative setting, then it is $\sum_{i=1}^{k} (v_i + \delta_i t - s_p d_i)$. The contribution in

the original setting is $\sum_{i \in K}(v_i + \delta_i t - s_p d_i)$, where K is a suitably chosen set of customers. The difference is:

$$\sum_{i \in [k] \setminus K} (v_i + \delta_i t - s_p d_i) - \sum_{i \in K \setminus [k]} (v_i + \delta_i t - s_p d_i) \leq 0. \tag{3.23}$$

The inequality holds because in the original setting at time t, customers in $[k] \setminus K$ are not contributing to p, while customers in $K \setminus [k]$ are, and hence

$$v_i + \delta_i t - s_p d_i \leq 0, \qquad i \in [k] \setminus K, \tag{3.24a}$$

$$v_i + \delta_i t - s_p d_i \geq 0, \qquad i \in K \setminus [k]. \tag{3.24b}$$

Therefore the contribution in the original setting is greater than or equal to the contribution in the alternative setting. (When $K = [k]$, the inequality holds with equality. When $K \neq [k]$, the inequality holds in the strict sense. The $\leq$ inequality suffices for this proof.)

Using Lemma 12, we can compute the first tangent to become tight in the alternative setting and the time when this occurs. Let the computed tangent and time be p'_{kl} and τ'_{kl}, and let t'_{kl} be the time in the original setting corresponding to time τ'_{kl} in the alternative setting. Then, as in Lemma 13, we can show that $t^* = \min\{t'_{kl} : l \in \{0, \ldots, L\}, k \in [m]\}$ and $p^* = p'_{\text{argmin}\{t'_{kl}: l \in \{0,\ldots,L\}, k \in [m]\}}$.

To evaluate the running time, note that given a time interval $[\theta_l, \theta_{l+1}]$, computing the values t'_{kl} and p'_{kl} for $k \in [m]$ requires $O(m)$ time. There are at most $m(m-1)/2$ orders, and computing them and the times when they change requires $O(m^2 \log m)$ time. Therefore, the total running time is $O(m^3)$.

To evaluate the number of ties, note that at time t^*, the points $p_i(t)$ divide $[0, D]$ into at most $m + 1$ intervals, and at most m of them contain tangents that are receiving customer contributions. By Lemma 12, there is at most one tight tangent on each of the m intervals, and thus if there is a tie, it can be between at most m tangents. □

3.3.3 Analysis of Multiple Facilities

We will now show how to execute an iteration of ALGORITHM FLPD in the case of multiple facilities. Recall that in Section 3.3.1, in addition to assuming the presence of only one facility, we had assumed that the connection costs c_{ij} were 0 and that there were no previously open tangents. We remove these assumptions as well.

In this section, we continue to refer to facilities of the infinitely-sized problem as tangents, and reserve the term facility for facilities of the concave cost problem. We say that customer i contributes to concave cost facility j if $v_i \geq c_{ij}$. We distinguish the situation when a customer contributes to concave cost facility j from the situation when a customer contributes to tangent p belonging to concave cost facility j.

Lemma 15. *An iteration of loop (2) of* ALGORITHM FLPD *can be executed in* $O(m^2 n)$ *time.*

Proof. First note that every time a tangent is opened, at least one unconnected customer is connected. Therefore, at any point in the algorithm, there are at most m open tangents. As a result, steps (4) and (5) can be performed in strongly polynomial time, since they only depend on the output from step (3), customers, and previously open tangents.

At the beginning of step (3), for each facility j, we define three events and associated times when they occur:

Time	Event
$t_1(j)$	A new tangent $\{j,p\}$ becomes tight.
$t_2(j)$	An unconnected customer i begins contributing to an open tangent $\{j,p\}$.
$t_3(j)$	An unconnected customer i begins contributing to concave facility j.

We begin by describing how to compute $\min\{t_1(j), t_2(j), t_3(j)\}$ under the assumption that no events at another facility j' occur in the meantime.

First, we compute $t_1(j)$ assuming that events 2 and 3 do not occur in the meantime, and denote the result of this computation by $t'_1(j)$. Under this assumption, we can compute $t_1(j)$ in strongly polynomial time by applying Lemma 13. Since we are assuming that $t_1(j) < t_3(j)$, the set of customers contributing to facility j will not change until time $t_1(j)$. Therefore, we can satisfy the lemma's assumption that $c_{ij} = 0$, by performing a coefficient transformation where we subtract c_{ij} from each v_i having $v_i \geq c_{ij}$. Since $t_1(j) < t_2(j)$, the lemma's assumption that a new tangent will become tight before an unconnected customer begins contributing to a previously open tangent is also satisfied.

Now consider the case when event 2 or 3 occurs first. We apply Lemma 13 in the same way, and distinguish the following cases.

1. If $t_1(j) > t_3(j)$ and $t_1(j) \leq t_2(j)$, a customer i begins contributing to facility j before a new tangent becomes tight. Yet the computation in the lemma is done as if i does not contribute to j at any point in time, and therefore $t'_1(j) > t_3(j)$.

2. If $t_1(j) > t_2(j)$ and $t_1(j) \leq t_3(j)$, the computation in the lemma will be done as if tangent $\{j,p\}$ is not open. Since we know that no tangents become tight before i begins contributing to $\{j,p\}$, we will have $t'_1(j) > t_2(j)$.

3. Similarly, if $t_1(j) > t_3(j)$ and $t_1(j) > t_2(j)$, then $t'_1(j) > \min\{t_3(j), t_2(j)\}$.

Next, we compute $t_2(j)$ assuming that events 1 and 3 do not occur in the meantime, and denote the result of the computation by $t'_2(j)$. Under this assumption, $t_2(j)$ can be computed easily—simply visit the list of open tangents, compute the time until each customer begins contributing to each tangent under the assumption that no other events intervene, and then take the minimum. If event 1 or 3 occurs before event 2, note that $t'_2(j) > \min\{t_1(j), t_3(j)\}$.

Similarly, assuming events 1 and 2 do not occur in the meantime, $t_3(j)$ can be computed easily, and we denote the result by $t'_3(j)$. If event 1 or 2 does occur before event 3, $t'_3(j) > \min\{t_1(j), t_2(j)\}$.

Therefore, when events at other facilities do not occur in the meantime, we have $\min\{t_1(j), t_2(j), t_3(j)\} = \min\{t'_1(j), t'_2(j), t'_3(j)\}$. If an event at another facility does occur

first at a time θ, then $\min\{t'_1(j), t'_2(j), t'_3(j)\} > \theta$. Therefore, $\min_j \min\{t_1(j), t_2(j), t_3(j)\} = \min_j \min\{t'_1(j), t'_2(j), t'_3(j)\}$.

Event 3 can occur at most mn times before events 1 or 2 occur. To perform step (3), we repeat the computation of $\min_j \min\{t'_1(j), t'_2(j), t'_3(j)\}$ until event 1 or 2 occurs.

To evaluate the running time, note the following:

1. The initial computation of $t'_1(j)$ for $j \in [n]$ requires $O(m^2)$ time per facility, and thus $O(m^2 n)$ time overall.

2. The initial computation of $t'_2(j)$ requires $O(1)$ time given a customer and open tangent. There are at most m open tangents, thus this step requires $O(m^2)$ time overall.

3. The initial computation and sorting of all $t'_3(j)$ requires $O(mn \log(mn))$ time.

4. Every time event 3 occurs involving a facility j, we have to update $t'_1(j)$, which requires $O(m)$ time per update, since we have to add only one new customer to the setting of Lemma 13. Event 3 occurs at most mn times, so the total time for all updates is $O(m^2 n)$.

Therefore, the total running time for one iteration is $O(m^2 n + mn \log n)$. □

We now easily obtain the running time for the overall algorithm.

Theorem 5. ALGORITHM FLPD *is a 1.61 approximation algorithm for concave cost facility location, with a running time of* $O(m^3 n + mn \log n)$.

Proof. Each iteration of loop (2) requires $O(mn \log(mn))$ time for sorting the times for event 3, and $O(m^2 n)$ for all other steps. However, the times for event 3 can be computed only once at the beginning of the algorithm. At each iteration of loop (2), an unconnected customer becomes connected, and hence there are at most m such iterations. Therefore, the total running time is $O(m^3 n + mn \log n)$.

By Theorem 4, due to Mahdian et al. [JMM+03], the algorithm for the fixed-charge problem is a 1.61 approximation algorithm. The approximation ratio follows directly from the fact that the algorithm for the concave cost problem mirrors the execution of the algorithm for the infinitely-sized fixed-charge problem. □

By a similar application of our technique to the 1.861 approximation algorithm of Mahdian et al. [JMM+03], we obtain a 1.861 approximation algorithm for concave cost facility location with a running time of $O(m^2 n + mn \log n)$.

3.4 Lot-Sizing with Concave Ordering Costs

In this section, we will apply the technique developed in Section 3.3 to lot-sizing. The lot-sizing problem with concave ordering costs is defined in Section 1.3:

$$\min \sum_{s=1}^{n} \phi_s \left(\sum_{t=s}^{n} d_t x_{st} \right) + \sum_{s=1}^{n} \sum_{t=s}^{n} h_{st} d_t x_{st}, \tag{3.25a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st} = 1, \qquad 1 \le t \le n, \tag{3.25b}$$

$$x_{st} \ge 0, \qquad 1 \le s \le t \le n. \tag{3.25c}$$

The meaning of coefficients and variables is the same as in Section 1.3. Recall that the problem is specified in terms of holding costs h_t, and we defined $h_{st} = \sum_{i=s}^{t-1} h_i$ for convenience. We continue to assume that the cost functions $\phi_t : \mathbb{R}_+ \to \mathbb{R}_+$ are concave and nondecreasing, and that the holding costs h_t and demands d_t are nonnegative.

Without loss of generality, we assume that all coefficients h_t and d_t are integral. Let $D = \sum_{t=1}^{n} d_t$ be the total demand.

The classical lot-sizing problem, also defined in Section 1.3, is given by:

$$\min \sum_{s=1}^{n} f_s y_s + \sum_{s=1}^{n} \sum_{t=s}^{n} (c_s + h_{st}) d_t x_{st}, \tag{3.26a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st} = 1, \qquad 1 \le t \le n, \tag{3.26b}$$

$$0 \le x_{st} \le y_s, \qquad 1 \le s \le t \le n. \tag{3.26c}$$

Note that we omit the constraints $y_{sp} \in \{0, 1\}$ as they are automatically satisfied at any extreme point solution [e.g. KB77].

We now proceed to develop an exact primal-dual algorithm for the concave cost lot-sizing problem, based on the algorithm of Levi et al. [LRS06] for classical lot-sizing. Levi et al. derive their algorithm in a slightly different setting, where the costs h_{st} are not necessarily the sum of per-period holding costs h_t, but rather satisfy an additional monotonicity condition. Here we adapt their algorithm in a natural way to work in the setting of problem (3.26).

The dual of problem (3.26) is given by:

$$\max \sum_{t=1}^{n} v_t, \tag{3.27a}$$

$$\text{s.t. } v_t \leq (c_s + h_{st})d_t + w_{st}, \qquad 1 \leq s \leq t \leq n, \tag{3.27b}$$

$$\sum_{t=s}^{n} w_{st} \leq f_s, \qquad 1 \leq s \leq n, \tag{3.27c}$$

$$w_{st} \geq 0, \qquad 1 \leq s \leq t \leq n. \tag{3.27d}$$

As with facility location, since the variables w_{st} are not present in the objective, we can assume that they are as small as is possible without violating constraint (3.27b). This yields the invariant $w_{st} = \max\{0, v_t - (c_s + h_{st})d_t\}$. Note that lot-sizing orders correspond to facilities in the facility location problem, and lot-sizing demand points correspond to customers in the facility location problem.

We refer to dual variable v_t as the *budget* of demand point t. If $v_t \geq (c_s + h_{st})d_t$, we say that demand point t *contributes* to the fixed cost f_s of order s, and w_{st} is its contribution. An order t is *tight* if its fixed charge is fully covered by demand point contributions, i.e. $\sum_{t=s}^{n} w_{st} = f_s$. An order is *over-tight* if $\sum_{t=s}^{n} w_{st} > f_s$.

Let (x, y) be an integral primal feasible solution, and (v, w) be a dual feasible solution. The primal complementary slackness constraints are:

$$x_{st}(v_t - (c_s + h_{st})d_t - w_{st}) = 0, \qquad 1 \leq s \leq t \leq n, \tag{3.28a}$$

$$y_s \left(\sum_{t=s}^{n} w_{st} - f_s \right) = 0, \qquad 1 \leq s \leq n. \tag{3.28b}$$

Constraint (3.28a) says that demand point t can be served from order s in the primal solution only if s is the closest to t with respect to the modified costs $c_s + h_{st} + w_{st}/d_t$. Constraint (3.28b) says that order t can be placed in the primal solution only if it is tight in the dual solution.

The algorithm of Levi et al. starts with dual feasible solution $(v, w) = 0$ and iteratively updates it, while maintaining dual feasibility and increasing the dual objective. At the same time, guided by the complementary slackness constraints, the algorithm constructs an integral primal solution. The algorithm concludes when the integral primal solution becomes feasible. An additional postprocessing step decreases the cost of the primal solution to the point where it equals that of the dual solution. At this point, the algorithm has computed an optimal solution to the lot-sizing problem.

We introduce the notion of a wave, which corresponds to the notion of time in the primal-dual algorithm for facility location. We will denote the wave position by W, and it will decrease continuously from h_{1n} to 0 and then possibly to a negative value not less than $-c_1 - f_1$. We associate to each step of the algorithm the wave position when it occurred.

ALGORITHM LSPD($n \in \mathbb{Z}_+$; $c, f, d \in \mathbb{R}^n_+, h \in \mathbb{R}^{n-1}_+$)

(1) Start with the wave at $W = h_{1n}$ and the dual solution $(v, w) = 0$. All orders are closed, and all demand points are unserved, i.e. $(x, y) = 0$.

(2) **While** there are unserved demand points:

(3) Decrease W continuously. At the same time increase v_t and w_{st} for unserved demand points t so as to maintain $v_t = \max\{0, d_t(h_{1t} - W)\}$ and $w_{st} = \max\{0, v_t - (c_s + h_{st})d_t\}$. The wave stops when an order becomes tight.

(4) Open the order s that became tight. For each unserved demand point t contributing to s, serve t from s.

(5) **For** each open order s from 1 to n:

(6) If there is a demand point t that contributes to s and to another open order s' with $s' < s$, close s. Reassign all demand points previously served from s to s'.

(7) Return (x, y) and (v, w).

In case of a tie between order points in step (4), we break the tie arbitrarily. Depending on the demand points that remain unserved, another one of the tied orders may open immediately in the next iteration of loop (2).

Theorem 6. ALGORITHM LSPD *is an exact algorithm for the lot-sizing problem.*

The proof is almost identical to that from [LRS06], and therefore for this proof we assume the reader is fully familiar with the lot-sizing results from [LRS06].

Proof. We will show that, after we have considered open order s, at the end of step (6), we maintain two invariants. First, each demand point is contributing to the fixed cost of at most one open order from the set $\{1, \ldots, s\}$. Second, each demand point is assigned to an open order and contributes to its fixed cost.

The first invariant follows from the definition of the algorithm. Indeed, if a demand point t' is contributing to s' and s with $s' < s$, then the algorithm would have closed s.

Clearly the second invariant holds at the beginning of loop (5). It continues to hold after we review order s if we have not closed s. Let us now consider the case when we have closed s. The demand points that have contributed to s can be classified into two categories. The first category contains the demand points whose dual variables stopped due to s becoming tight—these demand points were served from s and are now served from s'. Since t contributes to s', so do these demand points. The second category contains the demand points whose dual variables stopped due to another order s'' becoming tight. The case $s'' < s$ cannot happen, or s would have never opened. Hence, $s < s''$, and therefore s'' is currently open. Moreover, these demand points are currently served from s'' and are contributing to it.

Therefore, at the end of loop (5), each demand point is contributing to the fixed cost of at most one open order. Therefore, the fixed cost of opening orders is fully paid for by the dual solution. Moreover, each demand point is served from an open order, and therefore the primal solution is feasible. Since each demand point contributes to the fixed cost of the

order it is served from, the holding and variable connection cost is also fully paid for by the dual solution. Since the primal and dual solutions have the same cost, the algorithm is exact. □

The application of our technique to the lot-sizing problem is very similar to its application to the facility location problem in Section 3.3. Following that section, we reduce the concave cost lot-sizing problem (3.25) to the following infinitely-sized classical lot-sizing problem.

$$\min \sum_{s=1}^{n} \sum_{p\in[0,D]} f_{sp} y_{sp} + \sum_{s=1}^{n} \sum_{t=s}^{n} \sum_{p\in[0,D]} (c_{sp} + h_{st}) d_t x_{spt}, \tag{3.29a}$$

$$\text{s.t.} \sum_{s=1}^{t} \sum_{p\in[0,D]} x_{spt} = 1, \qquad 1 \le t \le n, \tag{3.29b}$$

$$0 \le x_{spt} \le y_{sp}, \qquad 1 \le s \le t \le n, p \in [0, D]. \tag{3.29c}$$

Again we note that since LP (3.29) is infinitely-sized, strongly duality does not hold automatically for it and its dual. However, in the proof of our algorithm we only rely on weak duality. Then, the fact that our algorithm produces a primal solution and a dual solution with the same cost proves that the solutions are optimal and that strong duality holds.

Theorem 7. ALGORITHM LSPD *is an exact algorithm for the concave cost lot-sizing problem, with a* $O(n^2)$ *running time.*

Proof. We define the following events and the wave positions when they occur:

Time	Event
$W_1(t)$	The wave reaches demand point t, i.e. $W = h_{1t}$.
$W_2(t)$	A tangent p of order point t becomes tight.

If for an order point t, no tangents become tight in the course of the algorithm, we set $W_2(t) = +\infty$. The wave positions $W_1(t)$ can be computed for all t at the beginning of the algorithm in $O(n)$. We compute the positions $W_2(t)$ by employing a set of intermediate values $W_2'(t)$.

We will show that each value $W_2'(t)$ will be the same as $W_2(t)$ on a truncated problem consisting of time periods $t, t+1, \ldots, n$. First, we compute $W_2'(n)$, which requires $O(1)$ time by Lemma 12. To compute $W_2'(t)$ given that $W_2'(t+1), \ldots, W_2'(n)$ are computed, we can employ Lemma 13.

The dual variables representing demand points $t, \ldots, n$ can be divided into three consecutive segments. First are the dual variables that are increasing at the same rate as part of the wave, then the dual variables v_k that are not increasing but exceed h_{tk}, and finally the dual variables v_k that are not increasing, do not exceed h_{tk}, and therefore have no role in this computation. We employ Lemma 13 and distinguish two cases:

1. Lemma 13 can be used to detect if a tangent is overtight. This indicates that $W_2'(t)$ is an earlier wave position than $W_2'(t+1), \ldots, W_2'(k)$ for some $k \leq n$. In this case, we delete $W_2'(t+1)$ from our list and repeat the computation of $W_2'(t)$ as if order point $t+1$ does not exist.

2. There are no overtight tangents. Thus, a tangent becomes tight at a wave position greater than or equal to $W_2'(t+1)$. In this case we set $W_2'(t)$ to this wave position, and proceed to the computation of $W_2'(t-1)$.

After computing $W_2'(t)$, consider the values that remain in our list and denote them by $W_2'(t), W_2'(\pi(1)), \ldots, W_2'(\pi(k))$ for some k. By induction, these values yield the correct times when tangents become tight for the truncated problem consisting of time periods $t, \ldots, n$. After we have computed $W_2'(1)$, the values $W_2'(t)$ remaining in our list yield the correct times $W_2(t)$, with the other values $W_2(t) = +\infty$. Therefore, loop (2) is complete.

A computation by Lemma 13 requires $O(n^2)$ time in the worst case. Since in this setting, all dual variables that are increasing exceed all dual variables that are stopped, each $W_2'(t)$ can be computed by Lemma 13 in $O(n)$. Each time we use Lemma 9 for a computation, a value $W_2'(t)$ is either removed from the list or inserted into the list. Since each value is inserted into the list only once, the total number of computations is $O(n)$, and the total running time for loop (2) is $O(n^2)$.

At the beginning of step (5), there are at most n open tangents, and n demand points, and therefore this loop can be implemented in $O(n^2)$ as well. □

Note that the values $W_2(t)$ also yield a dual optimal solution to the infinitely-sized LP. The solution can be computed from the $W_2(t)$-s in time $O(n)$ by taking $v_t = h_{1t} - W_2(\sigma(t))$, where $\sigma(t)$ is the latest time period less than or equal to t that has $W_2(\sigma(t)) < +\infty$.

3.5 Joint Replenishment with Concave Ordering Costs

In this section, we will apply our technique to the joint replenishment problem with concave individual ordering costs. This problem is defined in Section 1.4, and a mathematical programming formulation for it is given by:

$$\min \sum_{s=1}^{n} \phi^0 \left(\sum_{t=s}^{n} \sum_{k=1}^{K} d_t^k x_{st}^k \right) + \sum_{s=1}^{n} \sum_{k=1}^{K} \phi^k \left(\sum_{t=s}^{n} d_t^k x_{st}^k \right) + \sum_{s=1}^{n} \sum_{t=s}^{n} \sum_{k=1}^{K} h_{st}^k d_t^k x_{st}^k, \tag{3.30a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st}^k = 1, \qquad 1 \leq t \leq n, k \in [K], \tag{3.30b}$$

$$x_{st}^k \geq 0, \qquad 1 \leq s \leq t \leq n, k \in [K]. \tag{3.30c}$$

The meaning of coefficients and variables is as in Section 1.4. Note that the problem is given in terms of holding costs h_{st}^k, and we define $h_{st}^k = \sum_{i=s}^{t-1} h_i^k$ for convenience. As before, we assume that the cost functions $\phi^0 : \mathbb{R}_+ \to \mathbb{R}_+$ and $\phi^k : \mathbb{R}_+ \to \mathbb{R}_+$ are concave and

nondecreasing, and that the holding costs h_t^k and demands d_t^k are nonnegative. To reflect the fact that only the individual ordering costs are concave, the function ϕ^0 has the form $\phi^0(0) = 0$ and $\phi^0(x) = f^0$ for $x > 0$.

Without loss of generality, we assume that all coefficients h_t^k and d_t^k are integral. Let $D = \sum_{k=1}^{K} \sum_{t=1}^{n} d_t^k$ be the total demand.

Since the JRP with piecewise-linear concave ordering costs does not reduce to the classical JRP, we will develop our algorithm for the concave cost JRP as follows:

1. We consider a generalized JRP and an exponentially-sized formulation for it, as defined in Section 1.4.

2. We obtain a 4 approximation primal-dual algorithm for the generalized JRP based on this formulation, also with an exponential running time.

3. We reduce the concave cost JRP to an infinitely-sized generalized JRP. Then we use our technique to obtain a 4 approximation algorithm for the concave cost JRP with a strongly polynomial running time. As a byproduct we obtain a 4 approximation algorithm for the generalized JRP with a strongly polynomial running time.

Our technique was initially designed to obtain strongly polynomial algorithms based on formulations that were infinitely-sized due to a reduction from a concave cost problem. This application illustrates how it can be further used to obtain strongly polynomial algorithms based on formulations that are exponentially-sized due to the structure of the studied problem.

Let $\pi = (p_1, \ldots, p_K)$. A formulation for the generalized JRP is given by:

$$\min \sum_{\substack{s\in[n]\\ \pi\in[P]^K}} f^0 y_{s\pi}^0 + \sum_{\substack{s\in[n],k\in[K]\\ \pi\in[P]^K}} f_{p_k}^k y_{s\pi}^k + \sum_{\substack{1\le s\le t\le n\\ k\in[K],\pi\in[P]^K}} (c_{p_k}^k + h_{st}^k) d_t^k x_{s\pi t}^k, \tag{3.31a}$$

$$\text{s.t.} \sum_{\substack{s\in[t]\\ \pi\in[P]^K}} x_{s\pi t}^k = 1, \qquad 1 \le t \le n, k \in [K], \tag{3.31b}$$

$$0 \le x_{s\pi t}^k \le y_{s\pi}^0, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [P]^K, \tag{3.31c}$$

$$0 \le x_{s\pi t}^k \le y_{s\pi}^k, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [P]^K, \tag{3.31d}$$

$$y_{s\pi}^0 \in \{0,1\}, y_{s\pi}^k \in \{0,1\}, \qquad 1 \le s \le n, k \in [K], \pi \in [P]^K. \tag{3.31e}$$

As before, we assume that all costs and demands are nonnegative and integer. Note that a polynomially-sized formulation is possible for this problem, however it leads to a different primal dual algorithm, for which we are unable to obtain a constant factor approximation bound. Recall the intuition underlying the generalized JRP—at each time period, there are P options for placing an order for item k, and each option for an order of item k results in variable cost c_p^k and fixed cost f_p^k.

We will develop a 4 approximation algorithm for the generalized JRP based on the 2 approximation algorithm of Levi et al. [LRS06] for the classical JRP. We adapt their

algorithm in a natural way. Consider the LP relaxation of MIP (3.31) obtained by replacing the constraints $y^0_{s\pi} \in \{0,1\}, y^k_{s\pi} \in \{0,1\}$ with $y^0_{s\pi} \geq 0, y^k_{s\pi} \geq 0$. The dual of this LP relaxation is:

$$\max \sum_{k=1}^{K}\sum_{t=1}^{n} v_t^k, \tag{3.32a}$$

$$\text{s.t. } v_t^k \leq (c^k_{p_k} + h^k_{st})d^k_t + w^k_{s\pi t} + u^k_{s\pi t}, \qquad {\scriptstyle 1\leq s\leq t\leq n, k\in[K],\ \pi\in[P]^K,} \tag{3.32b}$$

$$\sum_{t=s}^{n} w^k_{s\pi t} \leq f^k_{p_k}, \qquad 1 \leq s \leq n, k \in [K], \pi \in [P]^K, \tag{3.32c}$$

$$\sum_{k=1}^{K}\sum_{t=s}^{n} u^k_{s\pi t} \leq f^0, \qquad 1 \leq s \leq n, \pi \in [P]^K, \tag{3.32d}$$

$$w^k_{s\pi t} \geq 0, u^k_{s\pi t} \geq 0, \qquad {\scriptstyle 1\leq s\leq t\leq n, k\in[K],\ \pi\in[P]^K.} \tag{3.32e}$$

Since now both $w^k_{s\pi t}$ and $u^k_{s\pi t}$ are not present in the objective, the invariants for them become more involved. When $\sum_{t=s}^{n} \max\{0, v^k_t - (c^k_{p_k} + h^k_{st})d^k_t\} \leq f^k_{p_k}$, we let as before

$$w^k_{s\pi t} = \max\{0, v^k_t - (c^k_{p_k} + h^k_{st})d^k_t\} \leq f^k_{p_k}. \tag{3.33a}$$

When $\sum_{t=s}^{n} \max\{0, v^k_t - (c^k_{p_k} + h^k_{st})d^k_t\} > f^k_{p_k}$, the algorithm will have previously fixed the values $w^k_{s\pi t}$ at the point when we had equality. In this situation, we let

$$u^k_{s\pi t} = \max\{0, v^k_t - (c^k_{p_k} + h^k_{st})d^k_t - w^k_{s\pi t}\}. \tag{3.33b}$$

We now have demand points for every time-item pair, and we refer to v^k_t as the *budget* of item k at time t. Given π, if $v^k_t \geq (c^k_{p_k} + h^k_{st})d_t$, we say that demand point (k,t) *contributes* to the fixed cost of item order (s,k,π) and let $w^k_{s\pi t}$ be its contribution. If $v^k_t \geq (c^k_{p_k} + h^k_{st})d_t$ and $\sum_{t=s}^{n} w^k_{s\pi t} = f^k_{p_k}$, we say that demand point (k,t) contributes to the fixed cost of joint order (s,π) and let $u^k_{s\pi t}$ be its contribution.

Since we now have several items, each with its own holding costs, we will think of W as a "master" wave, and decrease it from n to 0 and then to a bounded amount below 0. For each item k, we maintain an item wave

$$W^k = h_{1\lfloor W\rfloor} + h_{\lfloor W\rfloor}(W - \lfloor W\rfloor). \tag{3.34}$$

Intuitively, the W^k are computed so that the item waves arrive together at time periods $1, \ldots, n-1$ and advance linearly inbetween.

ALGORITHM JRP-PD($n, K, P \in \mathbb{Z}_+$; $f^0 \in \mathbb{Z}_+^P$; $f, c \in \mathbb{Z}_+^{KP}$;
$d \in \mathbb{Z}_+^{nK}$; $h \in \mathbb{Z}_+^{(n-1)K}$)

(1) Start with the wave at $W = n$ and the dual solution $(v, w, u) = 0$. All orders are closed, and all demand points are unserved, i.e. $(x, y) = 0$.
(2) **While** there are unserved demand points:
(3) Decrease W continuously and update W^k according to (3.34). At the same time, for unserved demand points (t, k), increase $v_t^k = \max\{0, d_t^k(h_{1t} - W^k)$, and update $w_{s\pi t}$ and $u_{s\pi t}$ so as to maintain (3.33). The wave stops when a joint or individual order becomes tight.
(4) If an individual order (s, k, π) became tight, fix the variables $w_{s\pi t}^k$ as described in (3.33). If the joint order (s, π) is tight, serve all demand points contributing to (s, k, π) from (s, π).
(5) If a joint order (s, π) became tight, open the joint order and all tight individual orders (s, π, k). For each unserved demand point (t, k) that contributes to joint order (s, π), serve (t, k) from (s, π).
(6) **For** each open joint order s from 1 to n:
(7) If there is a demand point t that contributes to s and to another open joint order s' with $s' < s$, close s.
(8) **For** each item k:
(9) **While** not all demand points have been processed in step (11):
(10) Select the latest such demand point (t, k). Let freeze(t, k) be the location of W^k when v_t^k was stopped, and let s be the earliest open joint order in $[\text{freeze}(t, k), t]$.
(11) Open item order (s, k). Serve all demand points (t', k) with $s \leq t' \leq t$ from (s, k).
(12) Return (x, y) and (v, w).

A direct implementation of this algorithm will have an exponential running time. It is possible to implement this algorithm to have a polynomial running time, however we will not do so here. Instead, we only prove that it is a 4 approximation algorithm.

Theorem 8. ALGORITHM JRP-PD *is a 4 approximation algorithm for the generalized joint replenishment problem.*

The proof closely resembles that from [LRS06], and therefore for this proof we assume the reader is fully familiar with the joint replenishment results from [LRS06].

Proof. First, similarly to the proof of Levi et al. and Theorem 6, after loop (6), each demand point contributes to at most one open joint order. Since we do not open any other joint orders after this step, the joint order cost is fully paid by the dual solution, i.e. $\sum_{s=1}^{n} \sum_{\pi \in [P]^K} f^0 y_{s\pi}^0 \leq \sum_{k=1}^{K} \sum_{s \in [n]} v_s^k$. Out of 4 times the cost of the dual solution, we allocate one toward the cost of the joint orders. Therefore, we need not consider the cost of the joint orders further in this proof.

Second, also similarly to the proof of Levi et al. and Theorem 6, after loop (6), for each demand point (t, k) there is at least one open joint order in $[\text{freeze}(t, k), t]$. Therefore, after loop (8), the algorithm produces a feasible primal solution.

Since we have already covered the cost of joint orders, we now consider each item k separately. We bound the holding cost and the cost of item orders in terms of the dual value, similarly to Levi et al. Due to the different cost structure of the JRP and generalized JRP, we are only able to bound the holding and item order cost by 3 times the cost of the dual solution, i.e. $\sum_{\substack{s\in[n]\\ \pi\in[P]^K}} f^k_{p_k} y^k_{s\pi} + \sum_{\substack{1\le s\le t\le n\\ \pi\in[P]^K}} (c^k_{p_k} + h^k_{st}) d^k_t x^k_{s\pi t} \le 3\sum_{t=1}^n v^k_t$.

Therefore, we obtain a 4 approximation algorithm. □

Finally, we obtain the strongly polynomial algorithm for the concave cost JRP. First, we reduce the concave cost JRP to an infinitely-sized generalized JRP. As before, we let $\pi = (p_1, \ldots, p_K)$, and the reduction is immediate:

$$\min \sum_{\substack{s\in[n]\\ \pi\in[0,D]^K}} f^0 y^0_{s\pi} + \sum_{\substack{s\in[n],k\in[K]\\ \pi\in[0,D]^K}} f^k_{p_k} y^k_{s\pi} + \sum_{\substack{1\le s\le t\le n\\ k\in[K]\\ \pi\in[0,D]^K}} (c^k_{p_k} + h^k_{st}) d^k_t x^k_{s\pi t}, \tag{3.35a}$$

$$\text{s.t.} \sum_{\substack{s\in[t]\\ \pi\in[0,D]^K}} x^k_{s\pi t} = 1, \qquad 1 \le t \le n, k \in [K], \tag{3.35b}$$

$$0 \le x^k_{s\pi t} \le y^0_{s\pi}, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [0,D]^K, \tag{3.35c}$$

$$0 \le x^k_{s\pi t} \le y^k_{s\pi}, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [0,D]^K, \tag{3.35d}$$

$$y^0_{s\pi} \in \{0,1\}, y^k_{s\pi} \in \{0,1\}, \qquad 1 \le s \le n, k \in [K], \pi \in [0,D]^K. \tag{3.35e}$$

We take the LP relaxation; the dual of the LP relaxation is:

$$\max \sum_{k=1}^K \sum_{t=1}^n v^k_t, \tag{3.36a}$$

$$\text{s.t. } v^k_t \le (c^k_{p_k} + h^k_{st}) d^k_t + w^k_{s\pi t} + u^k_{s\pi t}, \qquad {\scriptstyle 1\le s\le t\le n, k\in[K],\ \pi\in[0,D]^K,} \tag{3.36b}$$

$$\sum_{t=s}^n w^k_{s\pi t} \le f^k_{p_k}, \qquad 1 \le s \le n, k \in [K], \pi \in [0,D]^K, \tag{3.36c}$$

$$\sum_{k=1}^K \sum_{t=s}^n u^k_{s\pi t} \le f^0, \qquad 1 \le s \le n, \pi \in [0,D]^K, \tag{3.36d}$$

$$w^k_{s\pi t} \ge 0, u^k_{s\pi t} \ge 0, \qquad {\scriptstyle 1\le s\le t\le n, k\in[K],\ \pi\in[0,D]^K.} \tag{3.36e}$$

Theorem 9. ALGORITHM JRP-PD *is a 4 approximation algorithm for the concave cost joint replenishment problem, and can be executed in strongly polynomial time.*

Proof. Although in this setting all ordering costs are the same over time, we will need to refer to ordering costs and groups of tangents at specific times. With this in mind, we will refer to the ordering cost of item k at time t by (ϕ^k, t) and to the joint ordering cost at time t by (ϕ^0, t).

Note that we do not need to keep track of variables for each $\pi \in [0, D]^K$ explicitly. Denote the tangent to the item ordering cost (ϕ^k, s) that becomes tight first by p^*_{ks}. Then all the other tangents to this item ordering cost at this time are no longer relevant:

1. Concerning item ordering costs. For any item $l \neq k$, the behavior of demand points (t, l) or tangents to costs (ϕ^l, t) does not depend on item k, except through the joint ordering cost.

2. Concerning the joint ordering cost. For any wave position, the contribution to the joint ordering cost $\sum_{k=1}^{K} \sum_{t=s}^{n} u^k_{s\pi t}$ is highest for π with $p_k = p^*_{ks}$.

Therefore, it suffices to keep track, for each item k and time s, of the wave position when the first tangent to (ϕ^k, s) becomes tight. When this happens, we can stop considering all other tangents to (ϕ^k, s). When computing the wave position when the joint ordering cost becomes tight, we need to consider only the tangents that became tight for item ordering costs (ϕ^k, s). Through this transformation, the wave position when the joint ordering cost becomes tight can be computed by Lemma 13.

We now define the following events and wave positions when they occurred:

Wave Pos.	Event
$W_1(t)$	The wave reaches time period t, i.e. $W = t$.
$W_2(t, k)$	A tangent p of order point (t, k) becomes tight.
$W_3(t)$	The fixed joint ordering cost at period t becomes tight.

The computation now proceeds similarly to the lot-sizing case. We compute the largest of the wave positions $W_1(t)$, $W_2(t, k)$, and $W_3(t)$ (which corresponds to the smallest time in the facility location problem). After the computation we update the other W-values, and iterate. □

Chapter 4

Single Order Assignment

In this chapter, we consider the single order assignment problem—an operational problem arising at a leading internet retailer. The retailer operates a web-site through which it accepts customer orders, and several warehouses through which it fulfills these orders. An order consisting of multiple items arrives via the web-site and must be assigned to one or more of the warehouses, thereby resulting in one or more shipments. The assignment is subject to time, inventory availability, and other constraints, and the goal is to compute a feasible assignment that minimizes the fulfillment and transportation cost.

When formulated as an optimization problem, the single order assignment problem yields a problem that generalizes the facility location problem with non-metric connection costs, and therefore is NP-hard and does not admit a constant factor approximation algorithm unless P=NP [RS97]. We model this problem as a concave cost facility location problem, and employ existing primal-dual algorithms and approximations of piecewise-linear concave cost functions to solve it. On past data obtained from the retailer, our approach obtains solutions on average within 1.5% of optimality in less than 100ms per problem, even for the largest problems with over 240 items and 12 fulfillment centers.

Previously, researchers have evaluated the performance of algorithms for facility location using both randomly generated problems, and problems derived from practical applications. A number of algorithms, including primal-dual algorithms, have been found to perform near-optimally in these computational experiments [e.g. JMM$^+$03, BC05]. A survey of computational experiments for facility location is given by Hoefer [Hoe03].

To the best of our knowledge, our experiments differ from previous experiments on problems derived from practical applications in two ways. First, we employ facility location to solve a real-time operational problem, as opposed to the traditional use of facility location—making network design and facility placement decisions. Second, we employ primal-dual algorithms to solve problems with cost functions arising from shipping costs. These functions consist of many (≈ 100) pieces, and we approximate them with concave cost functions having on average less than 10 pieces. The algorithms maintain their near-optimal performance for this cost structure.

In Section 4.1, we describe the single order assignment problem. In Section 4.2, we model the problem as a concave cost facility location problem, and in Section 4.3, we present a summary of our computational results.

4.1 The Single Order Assignment Problem

The *single order assignment problem* is defined as follows. We are considering one customer order, consisting of a set of items $\{1, \ldots, m\}$. For each item $i \in [m]$, the customer has ordered d_i units, and has been promised shipment by a time τ_i. Note that in an order, different items may have different promise times.

The internet retailer has n fulfillment centers that can participate in fulfilling this order. Each center $j \in [n]$ has, for each item $i \in [m]$, an inventory availability list. For each item i we are given a set $T_{ij} = \{t_1, \ldots, t_{p_{ij}}\}$ of times when new inventory of item i becomes available at center j. In this set, t_1 denotes the present time, and $t_i \leq t_{i+1}$. We are further given a set of values b_{ijt} for $t \in T_{ij}$, indicating that b_{ijt} new units of item i will become available at center j at time t. The value b_{ijt_1} indicates the inventory of item i currently present at center j. For convenience, we also let $T_j = \cup_{i \in [m]} T_{ij}$ and $p_j = |T_j|$.

Moreover, for each fulfillment center we know whether it can ship to the given customer. Normally, there is a set of "core" fulfillment centers that can ship to most customers. Additional shipping options, such as direct shipping from a manufacturer, publisher, or distributor in response to an order on the retailer's website (known in the industry as *drop-shipping*) can be represented by additional fulfillment centers.

The shipping costs between a fulfillment center and a customer depend on several factors. One of them is the shipping method, which may represent different carriers, as well as different pricing options available at a carrier for different shipping matter. For example, the postal service provides special pricing for packages consisting only of books and packages consisting only of CDs. The shipping method is also affected by the shipping speed the customer selected—the speed may vary across items in the order, and across orders. Another factor is the geographical location of the fulfillment center and customer—the customer location is fixed for each instance, but will vary across problem instances.

Once these factors are fixed, the shipping costs are given by a list of pairs of the form (w_l, c_l), with each pair specifying that a shipment weighting w_l can be made at a cost of c_l. These cost pairs are given at intervals, generally of 0.1 pounds for weights from 0 to 1 pounds, and 1 pound for weights from 1 pound to 70–100 pounds. By linear interpolation, these weights form a shipping cost function.

The shipping cost functions are not concave, although they are concave on large weight intervals. We will approximate them with concave functions in Section 4.2, and then compare the optimal solutions with those obtained for the problem with the original cost functions in Section 4.3.

The following assumption is imposed by the retailer, and will enable us to model this problem as a facility location problem.

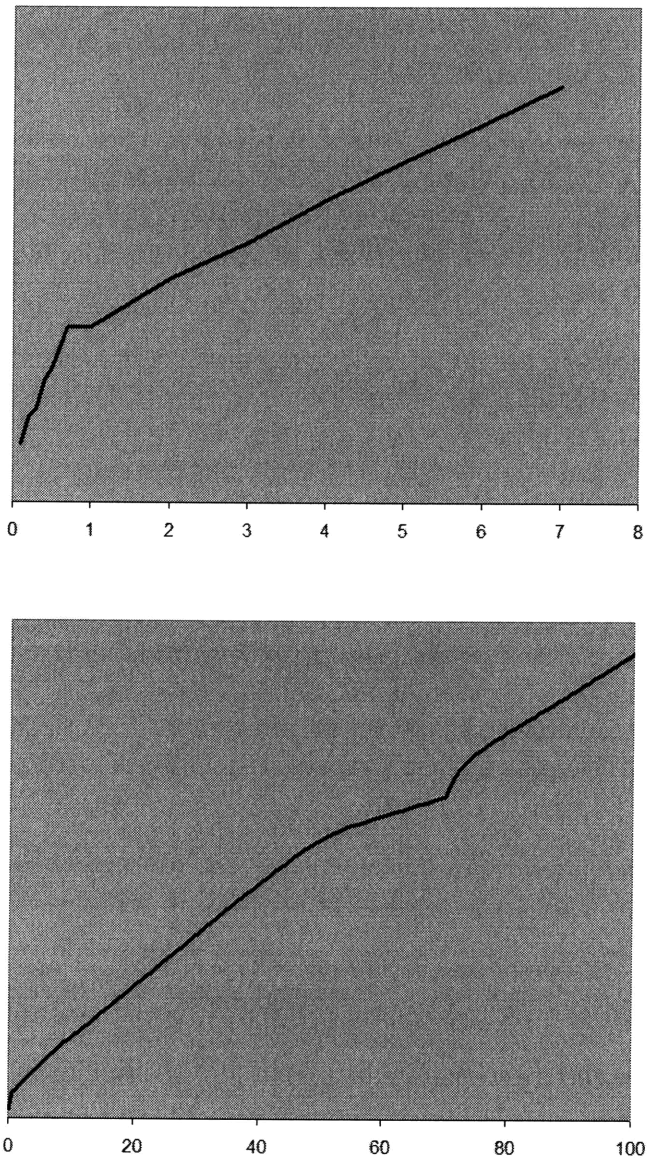

Figure 4-1: A sample cost function experienced by the retailer for a given shipping method from a given fulfillment center to a given customer. The horizontal scale represents the weight in lbs. The vertical scale represents the cost, however the units are omitted to exclude retailer confidential information.

Assumption 3. If a customer places an order for more than one unit of an item i, then all units of item i must ship from the same fulfillment center.

4.2 Concave Cost Models

First, we approximate the shipping cost functions by concave cost functions. Consider a cost function, for a specific shipping method, from a specific fulfillment center to the customer. A sample cost function is depicted in Figure 4-1. Note that, informally speaking, the function is not concave in two places—for very small weights, and for weights around 70 lbs.

Very heavy items, and in particular items heavier than $\approx$ 70 lbs, normally ship in a separate package. Therefore, we introduce a separate shipping method for heavy items and remove them from the shipping method under consideration. Doing so decreases the effect of the nonconcavity at $\approx$ 70 lbs.

We then eliminate the nonconcavities by replacing the cost functions by their upper concave envelopes. Finally, for all resulting cost functions, we reduce the number of pieces by considering all pieces from left to right and removing pieces as long as the resulting relative error for any given weight does not exceed 1%.

Of course, by modifying the cost function and then introducing a 1% approximation, we obtain a different model. For this reason, after performing our computations on the resulting model, we compare the cost of our solutions with the cost of optimal solutions or lower bounds computed with respect to the original, unmodified shipping cost functions.

Next, we model the problem as a concave cost multicommodity flow problem on a network as follows:

1. We represent the customer by a single node, and each ordered item by a commodity. Each fulfillment center $j \in [n]$ will be represented by a group of nodes (j, t) for times $t \in T_j$.

2. The supply of each item at each fulfillment center node (j, t) equals the number of new units of this item that become available at fulfillment center j at time t, or for the present time the number of units presently at the fulfillment center. That is, the supply equals b_{ijt}.

3. For each fulfillment center j, we consider all nodes $(j, t_1), (j, t_2), \ldots, (j, t_{p_j})$ and introduce inventory holding edges from (j, t_i) to (j, t_{i+1}).

We now construct the set of edges from the fulfillment center nodes to the customer node, and the set of commodities traveling on each edge to reflect the factors mentioned above—most importantly, time and shipping method:

1. If a shipping method only applies to a subset of items (say a shipping method that only allows books), then we allow only those commodities to flow on an edge.

2. If a shipping method does not permit the items to be shipped in time to the customer from the given fulfillment center node (j,t), then we omit that edge.

3. We create parallel edges between each fulfillment center node (j,t) and the customer node as needed to reflect the shipment methods available, and let the optimization routine decide which edge to use.

Lemma 16. *The single order assignment problem can be modeled as a concave cost facility location problem.*

Proof. First, we modify the concave cost multicommodity flow problem. We set the supply of each item i at each fulfillment center node (j,t) to the *total* inventory of item i available at center j at time t, i.e. the supply becomes $\sum_{s\in T_{ij},s\leq t} b_{ijs}$. Then, we delete the inventory holding edges. For each item $i \in A$, we remove the inventory of item i from fulfillment center nodes (j,t) where it is less than d_i. By Assumption 3, this formulation is equivalent to the original one.

It is now easy to construct an equivalent concave cost facility location formulation. We introduce a customer node for each *item*, and a facility for each edge. The facility concave cost function will be the same as the concave cost function on the corresponding edge. All customer connection costs will be zero. □

4.3 Computational Results

We have obtained from the retailer a problem set consisting of 60 of the largest orders received by the retailer within a given period of time. The orders contain up to 280 items, with an average of 71.34 items in an order. The orders can be served from up to 14 fulfillment centers, with an average of 11.64 fulfillment centers to be considered for an order. The largest order has 280 items and 14 fulfillment centers, and the second largest order has 240 items and 12 fulfillment centers.

These figures are before any processing or modeling is done on the single-order assignment problems. After processing and modeling, the resulting problems have up to 27 concave cost facilities, with an average of 13.70 facilities per problem. The concave cost functions have up to 18 pieces, with an average of 5.89 pieces per function. The largest order resulted in an instance with 13 concave cost facilities and 140 cost pieces in total (preprocessing eliminated one fulfillment center). The second largest order resulted in an instance with 18 concave cost facilities and 100 cost pieces in total.

Table 4.1 summarizes the computational results from solving the resulting facility location problems with ALGORITHM FLPD and a modified algorithm denoted by ALGORITHM FLPD-2. The modified algorithm is the same as the original one, except that in the modified algorithm the dual variables v_i do not change (decrease) after customer i connects to a facility.

The figures in columns FLPD and FLPD-2 denote the gaps obtained by ALGORITHM FLPD and ALGORITHM FLPD-2 on the concave cost facility location problems. The gaps

are computed with respect to the lower bounds provided by these algorithms. The figures in column "Best of two" represent the gap between the best of the two solutions and the best of the two lower bounds.

We then evaluated the cost of the two solutions with respect to the original concave functions, and computed an optimal solution or a lower bound to the problem with the original cost functions using CPLEX. The gap between the better of the two solutions and the lower bound or optimal solution computed by CPLEX is reported in column "Original fn.". The computational time is reported in column "CPU time".

	FLPD	FLPD-2	Best of two	Original fn.	CPU time
Max	8.59%	23.91%	6.12%	8.62%	**100ms**
Average	0.94%	1.57%	0.30%	**1.51%**	27ms

Table 4.1: Computational results for the single-order assignment problem.

In line "Max", we report the largest gaps and CPU times across all instances, while in line "Average", we report the average gaps and CPU times. Note that the average gap with respect to the original concave functions was 1.51%, and the maximum CPU time was 100ms. We conclude that the proposed approach produces near-optimal solutions to these instances, with very short running times even in the case of the largest instances.

Chapter 5

Conclusions

In this thesis, we studied the problem of minimizing a separable concave function over a polyhedron. Problems that fit into this framework arise often in practice, and have applications in transportation, logistics, telecommunications, and supply chain management. Specific problems that belong to this class include concave cost facility location, concave cost multicommodity flow, concave cost lot-sizing, and concave cost joint joint replenishment.

In Chapter 2 we introduce a general technique for approximating a concave cost problem by a piecewise linear problem. Our technique implies that polynomial-time heuristics, approximation algorithms, and exact algorithms for many discrete optimization problems immediately yield polynomial-time heuristics, approximation algorithms, and fully polynomial-time approximation schemes for the corresponding concave cost problems. We have illustrated this technique by obtaining a new approximation algorithm for concave cost facility location, and a new heuristic for a class of large-scale concave cost multicommodity flow problems.

A key result underlying the technique in Chapter 2 says that we can approximate any concave function that is nonnegative nondecreasing on $\mathbb{R}_+$ by a piecewise-linear function that will ensure a $1+\epsilon$ approximation on $[1, U]$ and will consist of $1+\lceil \log_{1+4\epsilon+4\epsilon^2} U \rceil$ pieces; as $\epsilon \to 0$, the number of pieces behaves as $1 + \frac{1}{4\epsilon} \log U$. We also establish a lower bound of $O\left(\frac{1}{\sqrt{\epsilon}} \log U\right)$ on the number of needed pieces. Bridging the gap between our technique and the lower bound is an interesting open question arising from Chapter 2.

In Chapter 3 we introduce an algorithm design technique that yields a strongly polynomial primal-dual algorithm for a concave cost problem whenever such an algorithm exists for the corresponding fixed-charge problem. The resulting algorithms operate directly on the concave cost problems and therefore are applicable in certain settings when piecewise-linear approaches are not. The approach preserves constant-factor approximation ratios, as well as certain non-constant ratios, and exact algorithms yield exact algorithms. We illustrate this technique by obtaining new approximation algorithms for concave cost facility location and concave cost joint replenishment, and a new exact algorithm for concave cost lot-sizing.

The technique of Chapter 3 enables us to obtain algorithms that operate directly on concave cost problems based on *primal-dual* algorithms for combinatorial optimization prob-

lems. Another technique that is prominent in the development of approximation algorithms is that of LP rounding. An interesting open question arising out of Chapter 3 is to develop a technique for obtaining algorithms that operate directly on concave cost problems based on LP rounding algorithms for combinatorial optimization problems.

In Chapter 4 we develop a solution method for an order assignment problem arising from the operations of a leading internet retailer. We first approximate the problem by a concave cost facility location problem, and then solve this problem using primal-dual algorithms. On past data provided by the retailer, our approach produced solutions that are on average within 1.5% of optimality in less than 100ms per problem, even for the largest problems in the data set. We conclude that the concave cost facility location model provides a close approximation for the original problem, and the primal-dual algorithms we have used are able to solve the resulting problems near-optimally within a short period of time.

Bibliography

[AE88] Y Askoy and S. S. Erenguk. Multi-item inventory models with coordinated replenishment. *Internat. J. Oper. Production Management*, 8:63–73, 1988.

[AJR89] Esther Arkin, Dev Joneja, and Robin Roundy. Computational complexity of uncapacited multi-echelon production planning problems. *Oper. Res. Lett.*, 8(2):61–66, 1989.

[And04] Matthew Andrews. Hardness of buy-at-bulk network design. In *FOCS '04: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS'04)*, pages 115–124, Washington, DC, USA, 2004. IEEE Computer Society.

[AP93] Alok Aggarwal and James K. Park. Improved algorithms for economic lot size problems. *Oper. Res.*, 41(3):549–571, 1993.

[Ata01] Alper Atamtürk. Flow pack facets of the single node fixed-charge flow polytope. *Oper. Res. Lett.*, 29(3):107–114, 2001.

[Bal66] M. L. Balinski. On finding integer solutions to linear programs. In *Proc. IBM Sci. Comput. Sympos. Combinatorial Problems (Yorktown Heights, N.Y., 1964)*, pages 225–248. IBM Data Process. Division, White Plains, N.Y., 1966.

[Bau58] Heinz Bauer. Minimalstellen von Funktionen und Extremalpunkte. *Arch. Math.*, 9:389–393, 1958.

[BC05] Francisco Barahona and Fabián A. Chudak. Near-optimal solutions to large-scale facility location problems. *Discrete Optim.*, 2(1):35–50, 2005.

[BH96] Cuneyt F. Bazlamacci and Khalil S. Hindi. Enhanced adjacent extreme-point search and tabu search for the minimum concave-cost uncapacitated transshipment problem. *The Journal of the Operational Research Society*, 47(9):1150–1165, 1996.

[BL97] Gavin J. Bell and Bruce W. Lamar. Solution methods for nonconvex network flow problems. In *Network optimization (Gainesville, FL, 1996)*, volume 450 of

Lecture Notes in Econom. and Math. Systems, pages 32–50. Springer, Berlin, 1997.

[BMMN95] M. O. Ball, T. L. Magnanti, C. L. Monma, and G. L. Nemhauser, editors. *Network models*, volume 7 of *Handbooks in Operations Research and Management Science*. North-Holland Publishing Co., Amsterdam, 1995.

[BMW89] A. Balakrishnan, T. L. Magnanti, and R. T. Wong. A dual-ascent procedure for large-scale uncapacitated network design. *Oper. Res.*, 37(5):716–740, 1989.

[Byr07] Jaroslaw Byrka. An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 4627 of *Lecture Notes in Comput. Sci.*, pages 29–43. Springer, Berlin, 2007.

[CG99] Moses Charikar and Sudipto Guha. Improved combinatorial algorithms for the facility location and k-median problems. In *40th Annual Symposium on Foundations of Computer Science (New York, 1999)*, pages 378–388. IEEE Computer Soc., Los Alamitos, CA, 1999.

[CG05] Moses Charikar and Sudipto Guha. Improved combinatorial algorithms for facility location problems. *SIAM J. Comput.*, 34(4):803–824 (electronic), 2005.

[CGM03] Keely L. Croxton, Bernard Gendron, and Thomas L. Magnanti. A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems. *Management Science*, 49(9):1268–1273, 2003.

[CNW90] Gérard Cornuéjols, George L. Nemhauser, and Laurence A. Wolsey. The uncapacitated facility location problem. In *Discrete location theory*, Wiley-Intersci. Ser. Discrete Math. Optim., pages 119–171. Wiley, New York, 1990.

[dB01] Carl de Boor. *A practical guide to splines*, volume 27 of *Applied Mathematical Sciences*. Springer-Verlag, New York, revised edition, 2001.

[EGP69] G. D. Eppen, F. J. Gould, and B. P. Pashigian. Extensions of the planning horizon theorem in the dynamic lot size model. *Management Sci.*, 15:268–277, 1969.

[FG07] Dalila B. M. M. Fontes and José Fernando Gonçalves. Heuristic solutions for general concave minimum cost network flow problems. *Networks*, 50(1):67–76, 2007.

[FHC03] Dalila B. M. M. Fontes, Eleni Hadjiconstantinou, and Nicos Christofides. Upper bounds for single-source uncapacitated concave minimum-cost network flow problems. *Networks*, 41(4):221–228, 2003. Special issue in memory of Ernesto Q. V. Martins.

[FHC06a] Dalila B. M. M. Fontes, Eleni Hadjiconstantinou, and Nicos Christofides. A branch-and-bound algorithm for concave network flow problems. *J. Global Optim.*, 34(1):127–155, 2006.

[FHC06b] Dalila B. M. M. Fontes, Eleni Hadjiconstantinou, and Nicos Christofides. A dynamic programming approach for solving single-source uncapacitated concave minimum cost network flow problems. *European J. Oper. Res.*, 174(2):1205–1219, 2006.

[FLR66] I. Feldman, F. A. Lehrer, and T. L. Ray. Warehouse location under continuous economies of scale. *Management Sci.*, 12:670–684, 1966.

[FT91] A. Federgruen and M. Tzur. A simple forward algorithm to solve general dynamic lot sizing models with n periods in $O(n \log n)$ or $O(n)$ time. *Management Sci.*, 37:909–925, 1991.

[FT94] Awi Federgruen and Michal Tzur. The joint replenishment problem with time-varying costs and demands: efficient, asymptotic and ϵ-optimal solutions. *Oper. Res.*, 42(6):1067–1086, 1994.

[Geo77] Arthur M. Geoffrion. Objective function approximations in mathematical programming. *Math. Programming*, 13(1):23–37, 1977.

[GK99] Sudipto Guha and Samir Khuller. Greedy strikes back: improved facility location algorithms. *J. Algorithms*, 31(1):228–248, 1999.

[GLS93] Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2 of *Algorithms and Combinatorics*. Springer-Verlag, Berlin, second edition, 1993.

[GM94] F. Güder and J. G. Morris. Optimal objective function approximation for separable convex quadratic programming. *Math. Programming*, 67(1, Ser. A):133–142, 1994.

[GP90] G. M. Guisewite and P. M. Pardalos. Minimum concave-cost network flow problems: applications, complexity, and algorithms. *Ann. Oper. Res.*, 25(1-4):75–99, 1990. Computational methods in global optimization.

[GP91] G. M. Guisewite and P. M. Pardalos. Algorithms for the single-source uncapacitated minimum concave-cost network flow problem. *J. Global Optim.*, 1(3):245–265, 1991.

[GS79] G. Gallo and C. Sodini. Adjacent extreme flows and application to min concave cost flow problems. *Networks*, 9(2):95–121, 1979.

[GW97] M.X. Goemans and D.P. Williamson. The primal-dual method for approximation algorithms and its application to network design problems. In Dorit S. Hochbaum, editor, *Approximation algorithms for NP-hard problems*, chapter 4, pages 144–191. PWS Pub. Co., Boston, 1997.

[HH98] Kaj Holmberg and Johan Hellstrand. Solving the uncapacitated network design problem by a Lagrangean heuristic and branch-and-bound. *Oper. Res.*, 46(2):247–259, 1998.

[HMM03] M. T. Hajiaghayi, M. Mahdian, and V. S. Mirrokni. The facility location problem with general cost functions. *Networks*, 42(1):42–47, 2003.

[Hoc82] Dorit S. Hochbaum. Heuristics for the fixed cost median problem. *Math. Programming*, 22(2):148–162, 1982.

[Hoe03] Martin Hoefer. Experimental comparison of heuristic and approximation algorithms for uncapacitated facility location. In *Experimental and efficient algorithms*, volume 2647 of *Lecture Notes in Comput. Sci.*, pages 165–178. Springer, Berlin, 2003.

[HS89] Dorit S. Hochbaum and Arie Segev. Analysis of a flow problem with fixed charges. *Networks*, 19(3):291–312, 1989.

[JMM+03] Kamal Jain, Mohammad Mahdian, Evangelos Markakis, Amin Saberi, and Vijay V. Vazirani. Greedy facility location algorithms analyzed using dual fitting with factor-revealing LP. *J. ACM*, 50(6):795–824 (electronic), 2003.

[Jon87] Dev Joneja. Multi-echelon and joint replenishment production and distribution systems with non-stationary demands. Technical Report TR000731, Cornell University Operations Research and Industrial Engineering, March 1987.

[KB77] Jakob Krarup and Ole Bilde. Plant location, set covering and economic lot size: an $O(mn)$-algorithm for structured problems. In *Numerische Methoden bei Optimierungsaufgaben, Band 3 (Tagung, Oberwolfach, 1976)*, pages 155–180. Internat. Ser. Numer. Math., Vol. 36. Birkhäuser, Basel, 1977.

[KH63] Alfred A. Kuehn and Michael J. Hamburdger. A heuristic program for locating warehouses. *Management Sci.*, 9:643–666, 1963.

[Kon00] Spyros Kontogiorgis. Practical piecewise-linear approximation for monotropic optimization. *INFORMS J. Comput.*, 12(4):324–340, 2000.

[KV02] Bernhard Korte and Jens Vygen. *Combinatorial optimization*, volume 21 of *Algorithms and Combinatorics*. Springer-Verlag, Berlin, second edition, 2002. Theory and algorithms.

[LRS05] Retsef Levi, Robin Roundy, and David B. Shmoys. A constant approximation algorithm for the one-warehouse multi-retailer problem. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 365–374, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics.

[LRS06] Retsef Levi, Robin O. Roundy, and David B. Shmoys. Primal-dual algorithms for deterministic inventory problems. *Math. Oper. Res.*, 31(2):267–284, 2006.

[LS06] Retsef Levi and Maxim Sviridenko. ' multi-retailer problem. In *Approximation, randomization and combinatorial optimization*, volume 4110 of *Lecture Notes in Comput. Sci.*, pages 188–199. Springer, Berlin, 2006.

[Man58] A.S. Manne. Programming of economic lot sizes. *Management Sci.*, 4:115–135, 1958.

[MF90] Pitu B. Mirchandani and Richard L. Francis, editors. *Discrete location theory.* Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons Inc., New York, 1990. A Wiley-Interscience Publication.

[MMP00] Adam Meyerson, Kamesh Munagala, and Serge Plotkin. Cost-distance: two metric network design. In *41st Annual Symposium on Foundations of Computer Science (Redondo Beach, CA, 2000)*, pages 624–630. IEEE Comput. Soc. Press, Los Alamitos, CA, 2000.

[MP03] Mohammad Mahdian and Martin Pál. Universal facility location. In *Algorithms—ESA 2003*, volume 2832 of *Lecture Notes in Comput. Sci.*, pages 409–421. Springer, Berlin, 2003.

[Mun03] Kamesh Munagala. *Approximation algorithms for concave cost network flow problems.* PhD thesis, Stanford University, Department of Computer Science, March 2003.

[MW84] T. L. Magnanti and R. T. Wong. Network design and transportation planning: Models and algorithms. *Trans. Sci.*, 18:1–55, 1984.

[MYZ06] Mohammad Mahdian, Yinyu Ye, and Jiawei Zhang. Approximation algorithms for metric facility location problems. *SIAM J. Comput.*, 36(2):411–432 (electronic), 2006.

[NW99] George Nemhauser and Laurence Wolsey. *Integer and combinatorial optimization.* Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons Inc., New York, 1999. Reprint of the 1988 original, A Wiley-Interscience Publication.

[OW03] Francisco Ortega and Laurence A. Wolsey. A branch-and-cut algorithm for the single-commodity, uncapacitated, fixed-charge network flow problem. *Networks*, 41(3):143–158, 2003.

[PK90] P. M. Pardalos and N. Kovoor. An algorithm for a singly constrained class of quadratic programs subject to upper and lower bounds. *Math. Programming*, 46(3, (Ser. A)):321–328, 1990.

[PR86] P. M. Pardalos and J. B. Rosen. Methods for global concave minimization: a bibliographic survey. *SIAM Rev.*, 28(3):367–379, 1986.

[RP86] J. B. Rosen and P. M. Pardalos. Global minimization of large-scale constrained concave quadratic problems by separable programming. *Math. Programming*, 34(2):163–174, 1986.

[RS97] Ran Raz and Shmuel Safra. A sub-constant error-probability low-degree test, and a sub-constant error-probability PCP characterization of NP. In *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 475–484, New York, NY, USA, 1997. ACM.

[RSSZ07] H. Edwin Romeijn, Thomas C. Sharkey, Zuo-Jun Max Shen, and Jiawei Zhang. Integrating facility location and production planning decisions. Preprint, 2007.

[Sch03] Alexander Schrijver. *Combinatorial optimization. Polyhedra and efficiency. Vol. A*, volume 24 of *Algorithms and Combinatorics*. Springer-Verlag, Berlin, 2003. Paths, flows, matchings, Chapters 1–38.

[SETA97] David B. Shmoys, Éva Tardos, and Karen Aardal. Approximation algorithms for facility location problems (extended abstract). In *STOC '97: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 265–274, New York, NY, USA, 1997. ACM.

[SS98] J. Parker Shectman and Nikolaos V. Sahinidis. A finite algorithm for global minimization of separable concave programs. *J. Global Optim.*, 12(1):1–35, 1998.

[SSLT] Zuo-Jun Shen, David Simchi-Levi, and Chung-Piaw Teo. Approximation algorithms for the single-warehouse multiretailer problem with piecewise linear cost structures. URL: citeseer.nj.nec.com/439759.html.

[Sto63] J. F. Stollsteimer. A working model for plant numbers and locations. *J. Farm Econom.*, 45:631–645, 1963.

[Svi02] Maxim Sviridenko. An improved approximation algorithm for the metric uncapacitated facility location problem. In *Integer programming and combinatorial optimization*, volume 2337 of *Lecture Notes in Comput. Sci.*, pages 240–257. Springer, Berlin, 2002.

[Tha78] Lakshman S. Thakur. Error analysis for convex separable programs: the piecewise linear approximation and the bounds on the optimal objective value. *SIAM J. Appl. Math.*, 34(4):704–714, 1978.

[Vei69] Arthur F. Veinott, Jr. Minimum concave-cost solution of Leontief substitution models of multi-facility inventory systems. *Operations Res.*, 17:262–291, 1969.

[Wag60] H. M. Wagner. A postscript to "dynamic problems in the theory of the firm". *Naval Res. Logist. Quart.*, 7:7–12, 1960.

[WvHK92] Albert Wagelmans, Stan van Hoesel, and Antoon Kolen. Economic lot sizing: an $O(n \log n)$ algorithm that runs in linear time in the Wagner-Whitin case. *Oper. Res.*, 40(suppl. 1):S145–S156, 1992.

[WW58] Harvey M. Wagner and Thomson M. Whitin. Dynamic version of the economic lot size model. *Management Sci.*, 5:89–96, 1958.

[Zab64] E. Zabel. Some generalizations of an inventory planning horizon theorem. *Management Sci.*, 10:465–471, 1964.

[Zan66a] W. I. Zangwill. A deterministic multi-period production scheduling model with backlogging. *Management Sci.*, 13:105–119, 1966.

[Zan66b] W. I. Zangwill. A deterministic multi-product, multi facility production and inventory model. *Operations Research*, 5:89–96, 1966.

[Zan68] W. I. Zangwill. Minimum concave cost flows in certain networks. *Management Sci.*, 14:429–450, 1968.