RUTCOR
RESEARCH
REPORT

Strongly Polynomial Primal-Dual Algorithms for Concave Cost Combinatorial Optimization Problems

Thomas L. Magnanti[a] Dan Stratila[b]

RRR 10-2012, February 2012

RUTCOR
Rutgers Center for
Operations Research
Rutgers University
640 Bartholomew Road
Piscataway, New Jersey
08854-8003
Telephone: 732-445-3804
Telefax: 732-445-5472
Email: rrr@rutcor.rutgers.edu
http://rutcor.rutgers.edu/∼rrr

[a]School of Engineering and Sloan School of Management, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Room 32-D784, Cambridge, MA 02139. E-mail: magnanti@mit.edu.

[b]Rutgers Center for Operations Research and Rutgers Business School, Rutgers University, 640 Bartholomew Road, Room 107, Piscataway, NJ 08854. E-mail: dstrat@rci.rutgers.edu.

Strongly Polynomial Primal-Dual Algorithms for Concave Cost Combinatorial Optimization Problems*

Thomas L. Magnanti† Dan Stratila‡

February 13, 2012

Abstract

We introduce an algorithm design technique for a class of combinatorial optimization problems with concave costs. This technique yields a strongly polynomial primal-dual algorithm for a concave cost problem whenever such an algorithm exists for the fixed-charge counterpart of the problem. For many practical concave cost problems, the fixed-charge counterpart is a well-studied 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.

Using our technique, we obtain a new 1.61-approximation algorithm for the concave cost facility location problem. For inventory problems, we obtain a new exact algorithm for the economic lot-sizing problem with general concave ordering costs, and a 4-approximation algorithm for the joint replenishment problem with general concave individual ordering costs.

1 Introduction

We introduce a general technique for designing strongly polynomial primal-dual algorithms for a class of combinatorial optimization problems with concave costs. We apply the technique to study three such problems: the concave cost facility location problem, the economic lot-sizing problem with general concave ordering costs, and the joint replenishment problem with general concave individual ordering costs.

In the second author's Ph.D. thesis [Str08] (see also [MS12]), we developed a general approach for approximating an optimization problem with a separable concave objective by an optimization problem with a piecewise-linear objective and the same feasible set. When we are minimizing a nonnegative cost function over a polyhedron, and would like the resulting problem to provide a $1 + \epsilon$ approximation to the original problem in optimal cost, the size of the resulting problem is polynomial in the size of the original problem

*This research is based on the second author's Ph.D. thesis at the Massachusetts Institute of Technology [Str08].

†School of Engineering and Sloan School of Management, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Room 32-D784, Cambridge, MA 02139. E-mail: magnanti@mit.edu.

‡Rutgers Center for Operations Research and Rutgers Business School, Rutgers University, 640 Bartholomew Road, Room 107, Piscataway, NJ 08854. E-mail: dstrat@rci.rutgers.edu.

and linear in $1/\epsilon$. This bound implies that a variety of polynomial-time exact algorithms, approximation algorithms, and polynomial-time heuristics for combinatorial optimization problems immediately yield fully polynomial-time approximation schemes, approximation algorithms, and polynomial-time heuristics for the corresponding concave cost problems.

However, the piecewise-linear approach developed in [Str08] cannot fully address several difficulties involving concave cost combinatorial optimization problems. First, the approach adds a relative error of $1+\epsilon$ in optimal cost. For example, using the approach together with an exact algorithm for the classical lot-sizing problem, we can obtain a fully polynomial-time approximation scheme for the lot-sizing problem with general concave ordering costs. However, there are exact algorithms for lot-sizing with general concave ordering costs [e.g. Wag60, AP93], making fully polynomial-time approximation schemes of limited interest.

Second, suppose that we are computing near-optimal solutions to a concave cost problem by performing a $1 + \epsilon$ piecewise-linear approximation, and then using a heuristic for the resulting combinatorial optimization problem. We are facing a trade-off between choosing a larger value of ϵ and introducing an additional approximation error, or choosing a smaller value of ϵ and having to solve larger combinatorial optimization problems. For example, in [Str08], we computed near-optimal solutions to large-scale concave cost multicommodity flow problems by performing piecewise-linear approximations with $\epsilon = 1\%$, and then solving the resulting fixed-charge multicommodity flow problems with a primal-dual heuristic. The primal-dual heuristic itself yielded an average approximation guarantee of 3.24%. Since we chose $\epsilon = 1\%$, the overall approximation guarantee averaged 4.27%. If we were to choose $\epsilon = 0.1\%$ in an effort to lower the overall guarantee, the size of the resulting problems would increase by approximately a factor of 10.

Third, in some cases, after we approximate the concave cost problem by a piecewise-linear problem, the resulting problem does not reduce polynomially to the corresponding combinatorial optimization problem. As a result, the piecewise-linear approach in [Str08] cannot obtain fully polynomial-time approximation schemes, approximation algorithms, and polynomial-time heuristics for the concave cost problem. For example, when we carry out a piecewise-linear approximation of the joint replenishment problem with general concave individual ordering costs, the resulting joint replenishment problem with piecewise-linear individual ordering costs can be reduced only to an exponentially-sized classical joint replenishment problem.

These difficulties are inherent in any piecewise-linear approximation approach, and cannot be addressed fully without making use of the problem structure.

The technique developed in this paper yields a strongly polynomial primal-dual algorithm for a concave cost problem whenever such an algorithm exists for the corresponding combinatorial optimization problem. The resulting algorithm runs directly on the concave cost problem, yet can be viewed as the original algorithm running on an exponentially or infinitely-sized combinatorial optimization problem. Therefore, exact algorithms yield exact algorithms, and constant factor approximation ratios are preserved. Since the execution of the resulting algorithm mirrors that of the original algorithm, we can also expect the aposteriori approximation guarantees of heuristics to be similar in many cases.

1.1 Literature Review

1.1.1 Concave Cost Facility Location

In the *classical facility location* problem, there are m customers and n facilities. Each customer i has a demand $d_i > 0$, and needs to be connected to an open facility to satisfy this demand. Connecting a customer i to a facility j incurs a connection cost $c_{ij}d_i$; we assume that the connection costs are nonnegative and satisfy the metric inequality. Each facility j has an associated opening cost $f_j \in \mathbb{R}_+$. Let $x_{ij} = 1$ if customer i is connected to facility j, and $x_{ij} = 0$ otherwise. Also let $y_j = 1$ if facility j is open, and $y_j = 0$ otherwise. Then the total cost is $\sum_{j=1}^{n} f_j y_j + \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}d_i x_{ij}$. The goal is to assign each customer to one facility, while minimizing the total cost.

The classical facility location problem is one of the fundamental problems in operations research [CNW90, NW99]. The reference book edited by Mirchandani and Francis [MF90] introduces and reviews the literature for a number of location problems, including classical facility location. Since in this paper, our main contributions to facility location problems are in the area of approximation algorithms, we next provide a brief survey of previous approximation algorithms for classical facility location.

Hochbaum [Hoc82] showed that the greedy algorithm provides a $O(\log n)$ approximation for this problem, even when the connection costs c_{ij} are not metric. Shmoys et al. [STA97] gave the first constant-factor approximation algorithm, with a guarantee of 3.16. More recently, Jain et al. introduced primal-dual 1.861 and 1.61-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 complexity of approximation, the more general problem where the connection costs need not be metric has the set cover problem as a special case, and therefore is not approximable to within a certain logarithmic factor unless P = NP [RS97]. The problem with metric costs does not have a polynomial-time approximation scheme unless P = NP, and is not approximable to within a factor of 1.463 unless NP $\subseteq$ DTIME $\left(\eta^{O(\log\log\eta)}\right)$ [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 including a fixed charge f_j for opening each facility j. As one of the simplest forms of concave functions, fixed charge costs enable the model to capture the trade-off 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 j a nondecreasing concave cost function $\phi_j : \mathbb{R}_+ \to \mathbb{R}_+$, capturing a wider variety of phenomena than is possible with fixed charges. We assume without loss of generality that $\phi_j(0) = 0$ for all j. The cost at facility j is a function of the total demand at j, that is $\phi_j\left(\sum_{i=1}^{m} d_i x_{ij}\right)$, and the total cost is $\sum_{j=1}^{n} \phi_j\left(\sum_{i=1}^{m} d_i x_{ij}\right) + \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}d_i x_{ij}$.

Researchers have studied the concave cost facility location problem since at least the 1960's [KH63, FLR66]. Since it contains classical facility location as a special case, the

previously mentioned complexity results hold for this problem—the more general non-metric problem cannot be approximated to within a certain logarithmic factor unless P = NP, and the metric problem cannot be approximated to within a factor of 1.463 unless NP $\subseteq$ DTIME $\left(\eta^{O(\log\log\eta)}\right)$.

To the best of our knowledge, previously the only constant factor approximation algorithm for concave cost facility location was obtained by Mahdian and Pal [MP03], who developed a $3+\epsilon$ approximation algorithm based on local search.

When the concave cost facility location problem has uniform demands, that is $d_1 = d_2 = \cdots = d_m$, 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 concave cost facility location with uniform demands can be reduced polynomially to classical facility location. For example, Hajiaghayi et al. [HMM03] and Mahdian et al. [MYZ06] described a 1.52-approximation algorithm.

In the second author's Ph.D. thesis [Str08] (see also [MS12]), we obtain a $1.4991+\epsilon$ approximation algorithm for concave cost facility location by using piecewise-linear approximation. The running time of this algorithm depends polynomially on $1/\epsilon$; when ϵ is fixed, the running time is not strongly polynomial.

Independently, Romeijn et al. [RSSZ10] developed strongly polynomial 1.61 and 1.52-approximation algorithms for this problem, each with a running time of $O(n^4 \log n)$. Here n is the higher of the number of customers and the number of facilities. They consider the algorithms for classical facility location from [JMM$^+$03, MYZ06] through a greedy perspective. Since this paper uses a primal-dual perspective, establishing a connection between the research of Romeijn et al. and ours is an interesting question.

1.1.2 Concave Cost Lot-Sizing

In the *classical lot-sizing* problem, we have n discrete time periods, and a single item (sometimes referred to as a product, or commodity). In each time period $t = 1, \ldots, n$, there is a demand $d_t \in \mathbb{R}_+$ 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 at time t consists of a fixed cost $f_t \in \mathbb{R}_+$ and a per-unit cost $c_t \in \mathbb{R}_+$: ordering ξ_t units costs $f_t + c_t\xi_t$. Holding inventory from time t to time $t+1$ involves a per-unit holding cost $h_t \in \mathbb{R}_+$: holding ξ_t units costs $h_t\xi_t$. The goal is to satisfy all demand, while minimizing the total ordering and holding cost.

The classical lot-sizing problem is one of the basic problems in inventory management and was introduced by 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; for a broader overview, the reader may refer to the book by Pochet and Wolsey [PW06]. Wagner and Whitin [WW58] provided a $O(n^2)$ algorithm 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 the general case.

Krarup and Bilde [KB77] showed that integer programming formulation used in Section

3 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.)

The *concave cost lot-sizing* problem generalizes classical lot-sizing by replacing the fixed and per-unit ordering costs f_t and c_t with nondecreasing concave cost functions $\phi_t : \mathbb{R}_+ \to \mathbb{R}_+$. The cost of ordering ξ_t units at time t is now $\phi_t(\xi_t)$. We assume without loss of generality that $\phi_t(0) = 0$ for all t. 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 of $O(n^2)$.

1.1.3 Concave Cost Joint Replenishment

In the *classical joint replenishment* problem (JRP), we have n discrete time periods, and K items (which may also be referred to as products, or commodities). For each item k, the set-up is similar to the classical lot-sizing problem. There is a demand $d_t^k \in \mathbb{R}_+$ 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 $h_t^k \in \mathbb{R}_+$ for holding a unit of item k from time t to $t+1$. For each order of item k at time t, we incur a fixed cost $f^k \in \mathbb{R}_+$. Distinguishing the classical JRP from K separate classical lot-sizing problems is the fixed joint ordering cost—for each order at time t, we pay a fixed cost of $f^0 \in \mathbb{R}_+$, independent of the number of items or units ordered at time t. Note that f^0 and f^k do not depend on the time t. The goal is to satisfy all demand, while minimizing the total ordering and holding cost.

The classical JRP is a basic model in inventory theory [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 [Zan66, Vei69]. Federgruen and Tzur [FT94] developed a heuristic that computes $1+\epsilon$ approximate solutions provided certain input parameters are bounded. Shen et al [SSLT] obtained a $O(\log n + \log K)$ approximation algorithm for the one-warehouse multi-retailer problem, which has the classical 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 joint replenishment* problem generalizes the classical JRP by replacing the fixed individual ordering costs f^k by nondecreasing concave cost functions $\phi^k : \mathbb{R}_+ \to \mathbb{R}_+$. We assume without loss of generality that $\phi^k(0) = 0$ for all k. The methods employed by Zangwill [Zan66] and Veinott [Vei69] for the classical JRP with a fixed number of items or fixed number of time periods can also be employed on the concave cost JRP. 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], a constant factor approximation algorithm for the classical JRP was not known, we conclude that no constant factor approximation algorithms are known for the concave cost JRP.

1.2 Our Contribution

In Section 2, we develop our algorithm design technique for the concave cost facility location problem. In Section 2.1, we describe preliminary concepts. In Sections 2.2 and 2.3, we obtain the key technical insights on which our approach is based. In Section 2.4, we obtain a strongly polynomial 1.61-approximation algorithm for concave cost facility location with a running time of $O(m^3n + mn \log n)$. We can also obtain a strongly polynomial 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.

In Section 3, we apply our technique to the concave cost lot-sizing problem. We first adapt the algorithm of Levi et al. [LRS06] to work for the classical lot-sizing problem as defined in this paper. Levi et al. derive their algorithm in a slightly different setting that is neither a generalization nor a special case of the setting in this paper. In Section 3.1, we obtain a strongly polynomial exact algorithm for concave cost lot-sizing with a running time of $O(n^2)$. Here n is the number of time periods. While the running time matches that of the fastest previous algorithm [AP93], our main goal is to use this algorithm as a stepping stone in the development of our approximation algorithm for the concave cost JRP in the following section.

In Section 4, we apply our technique to the concave cost JRP. We first describe the difficulty in using piecewise-linear approximation on the concave cost JRP. We then introduce a more general version of the classical JRP, which we call generalized JRP, and an exponentially-sized integer programming formulation for it. In Section 4.1, using the 2-approximation algorithm of Levi et al. [LRS06] as the basis, we obtain an algorithm for the generalized JRP that provides a 4-approximation guarantee and has exponential running time. In Section 4.2, we obtain a strongly polynomial 4-approximation algorithm for the concave cost JRP.

2 Concave Cost Facility Location

We first develop our technique for concave cost facility location, and then apply it to other problems. We begin by describing the 1.61-approximation algorithm for classical facility location due to Jain et al. [JMM+03]. We assume the reader is familiar with the primal-dual method for approximation algorithms [see e.g. GW97].

Let $[n] = \{1, \ldots, n\}$. The classical facility location problem, defined in Section 1.1.1, can be formulated as an integer program as follows:

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

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

$$0 \leq x_{ij} \leq y_j, \qquad i \in [m], j \in [n], \tag{1c}$$

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

Recall that $f_j \in \mathbb{R}_+$ are the facility opening costs, $c_{ij} \in \mathbb{R}_+$ are the costs of connecting customers to facilities, and $d_i \in \mathbb{R}_+$ are the customer demands. We assume that the

connection costs c_{ij} obey the metric inequality, and that the demands d_i are positive. Note that we do not need the constrains $x_{ij} \in \{0,1\}$, since for any fixed $y \in \{0,1\}^n$, the resulting feasible polyhedron is integral.

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

$$\max \sum_{i=1}^{m} v_i, \tag{2a}$$

$$\text{s.t. } v_i \leq c_{ij} d_i + w_{ij}, \qquad i \in [m], j \in [n], \tag{2b}$$

$$\sum_{i=1}^{m} w_{ij} \leq f_j, \qquad j \in [n], \tag{2c}$$

$$w_{ij} \geq 0, \qquad i \in [m], j \in [n]. \tag{2d}$$

Since w_{ij} do not appear in the objective, we can assume that they are as small as possible without violating constraint (2b). 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 facility j, and w_{ij} is its contribution. The total contribution received by a facility j is $\sum_{i=1}^{m} w_{ij}$. A facility j is *tight* if $\sum_{i=1}^{m} w_{ij} = f_j$ and *over-tight* if $\sum_{i=1}^{m} w_{ij} > f_j$.

The primal complementary slackness constraints are:

$$x_{ij}(v_i - c_{ij} d_i - w_{ij}) = 0, \qquad i \in [m], j \in [n], \tag{3a}$$

$$y_j \left(\sum_{i=1}^{n} w_{ij} - f_j \right) = 0, \qquad j \in [n]. \tag{3b}$$

Suppose that (x, y) is an integral primal feasible solution, and (v, w) is a dual feasible solution. Then, constraint (3a) 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 (3b) 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 dual objective is not necessarily monotonic.) At the same time, guided by the primal 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 provides 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 occurs. In the algorithm, we denote the time 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 $t = 0$ 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 $v_i = td_i$ and $w_{ij} = \max\{0, v_i - c_{ij}d_i\}$. The increase stops when a closed facility becomes tight, or an unconnected customer begins contributing to an open facility.

(4) If a closed facility j became tight, open it. For each customer i that contributes to j, connect i to j, set $v_i = c_{ij}d_i$, and set $w_{ij'} = \max\{0, v_i - c_{ij'}d_i\}$ for all facilities j'.

(5) If an unconnected customer i began contributing to an open facility j, connect i to j.

(6) 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 1 (JMM+03)**.** *Algorithm* FLPD *is a 1.61-approximation algorithm for the classical facility location problem.*

Note that the integer program and the algorithm in our presentation are different from those in [JMM+03]. However both the integer program and the algorithm are equivalent to those in the original presentation.

2.1 The Technique

The concave cost facility location problem, also defined in Section 1.1.1, can be written as a mathematical program:

$$\min \sum_{j=1}^{n} \phi_j \left(\sum_{i=1}^{m} d_i x_{ij} \right) + \sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij} d_i x_{ij}, \tag{4a}$$

$$\text{s.t.} \sum_{j=1}^{n} x_{ij} = 1, \qquad i \in [m], \tag{4b}$$

$$x_{ij} \geq 0, \qquad i \in [m], j \in [n]. \tag{4c}$$

Here, $\phi_j : \mathbb{R}_+ \to \mathbb{R}_+$ are the facility cost functions, with each function being concave nondecreasing. Assume without loss of generality that $\phi_j(0) = 0$ for all cost functions. We omit the constraints $x_{ij} \in \{0, 1\}$, which are automatically satisfied at any vertex of the feasible polyhedron. Since the objective is concave, this problem always has a vertex optimal solution [HH61].

Suppose that the concave functions ϕ_j are piecewise linear on $(0, +\infty)$ with P pieces.

The functions can be written as

$$\phi_j(\xi_j) = \begin{cases} \min\{f_{jp} + s_{jp}\xi_j : p \in [P]\}, & \xi_j > 0, \\ 0, & \xi_j = 0. \end{cases} \tag{5}$$

As is well-known [e.g. FLR66], in this case problem (4) can be written as the following integer program:

$$\min \sum_{j=1}^{n}\sum_{p=1}^{P} f_{jp}y_{jp} + \sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{p=1}^{P}(c_{ij} + s_{jp})d_i x_{ijp}, \tag{6a}$$

$$\text{s.t.} \sum_{j=1}^{n}\sum_{p=1}^{P} x_{ijp} = 1, \qquad i \in [m], \tag{6b}$$

$$0 \le x_{ijp} \le y_{jp}, \qquad i \in [m], j \in [n], p \in [P], \tag{6c}$$

$$y_{jp} \in \{0, 1\}, \qquad j \in [n], p \in [P]. \tag{6d}$$

This 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 problem (4) corresponds to a facility $\{j, p\}$ in this problem. The new facility has opening cost f_{jp}. The set of customers is the same, and the connection cost from facility $\{j, p\}$ to customer i is $c_{ij} + s_{jp}$. Note that the new connection costs again satisfy the metric inequality.

We now return to the general case, when the functions ϕ_j need not be piecewise linear. Assume that ϕ_j are given by an oracle that returns the function value $\phi_j(\xi_j)$ and derivative $\phi'_j(\xi_j)$ in time $O(1)$ for $\xi_j > 0$. If the derivative at ξ_j does not exist, the oracle returns the right derivative, and we denote $\phi'_j(\xi_j) = \lim_{\zeta \to \xi_j +} \frac{\phi_j(\zeta) - \phi_j(\xi_j)}{\zeta - \xi_j}$. The right derivative always exists at $\xi_j > 0$, since ϕ_j is concave on $[0, +\infty)$.

We interpret each concave function ϕ_j as a piecewise-linear function with an infinite number of pieces. For each $p > 0$, we introduce a tangent $f_{jp} + s_{jp}\xi_j$ to ϕ_j at p, with

$$s_{jp} = \phi'_j(p), \qquad f_{jp} = \phi_j(p) - ps_{jp}. \tag{7}$$

We also introduce a tangent $f_{j0} + s_{j0}\xi_j$ to ϕ_j at 0, with $f_{j0} = \lim_{p \to 0+} f_{jp}$ and $s_{j0} = \lim_{p \to 0+} s_{jp}$. The limit $\lim_{p \to 0+} f_{jp}$ is finite because f_{jp} are nondecreasing in p and bounded from below. The limit $\lim_{p \to 0+} s_{jp}$ is either finite or $+\infty$ because s_{jp} are nonincreasing in p, and we assume that this limit is finite.

Our technique also applies when $\lim_{p \to 0+} s_{jp} = +\infty$, in which case we introduce tangents to ϕ_j only at points $p > 0$, and then proceed in similar fashion. In some computational settings, using derivatives is computationally expensive. In such cases, we can assume that the demands are rational, and let $d_i = \frac{d'_i}{d''_i}$ with $d''_i > 0$, and d'_i and d''_i coprime integers. Also let $\Delta = \frac{1}{d'_1 d'_2 \ldots d'_m}$. Then, we can use the quantity $\frac{\phi_j(\lfloor \xi_j + \Delta \rfloor) - \phi_j(\lfloor \xi_j \rfloor)}{\Delta}$ instead of $\phi'_j(\xi_j)$ throughout.

The functions ϕ_j can now be expressed as:

$$\phi_j(\xi_j) = \begin{cases} \min\{f_{jp} + s_{jp}\xi_j : p \ge 0\}, & \xi_j > 0, \\ 0, & \xi_j = 0. \end{cases} \tag{8}$$

When ϕ_j is linear on an interval $[\zeta_1, \zeta_2]$, all points $p \in [\zeta_1, \zeta_2)$ yield the same tangent, that is $(f_{jp}, s_{jp}) = (f_{jq}, s_{jq})$ for any $p, q \in [\zeta_1, \zeta_2)$. For convenience, we consider the tangents (f_{jp}, s_{jp}) for all $p \geq 0$, regardless of the shape of ϕ_j. Sometimes, we will refer to a tangent (f_{jp}, s_{jp}) by the point p that gave rise to it.

We apply formulation (6), and obtain a classical facility location problem with m customers and an infinite number of facilities. Each tangent p to cost function ϕ_j of facility j in problem (4) corresponds to a facility $\{j, p\}$ in the resulting problem. Due to their origin, we will sometimes refer to facilities in the resulting problem as tangents.

The resulting integer program is:

$$\min \sum_{j=1}^{n} \sum_{p \geq 0} f_{jp} y_{jp} + \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{p \geq 0} (c_{ij} + s_{jp}) d_i x_{ijp}, \tag{9a}$$

$$\text{s.t.} \sum_{j=1}^{n} \sum_{p \geq 0} x_{ijp} = 1, \qquad i \in [m], \tag{9b}$$

$$0 \leq x_{ijp} \leq y_{jp}, \qquad i \in [m], j \in [n], p \geq 0, \tag{9c}$$

$$y_{jp} \in \{0, 1\}, \qquad j \in [n], p \geq 0. \tag{9d}$$

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 this problem implicitly. Formally, we will devise an algorithm that takes problem (4) as input, runs in polynomial time, and produces the same assignment of customers to facilities as if Algorithm FLPD were run on problem (9). Thereby, we will obtain a 1.61-approximation algorithm for problem (4). We will call the new algorithm ConcaveFLPD.

The LP relaxation of problem (9) is obtained by replacing the constraints $y_{jp} \in \{0, 1\}$ with $y_{jp} \geq 0$. The dual of the LP relaxation is:

$$\max \sum_{i=1}^{m} v_i, \tag{10a}$$

$$\text{s.t. } v_i \leq (c_{ij} + s_{jp}) d_i + w_{ijp}, \qquad i \in [m], j \in [n], p \geq 0, \tag{10b}$$

$$\sum_{i=1}^{m} w_{ijp} \leq f_{jp}, \qquad j \in [n], p \geq 0, \tag{10c}$$

$$w_{ijp} \geq 0, \qquad i \in [m], j \in [n], p \geq 0. \tag{10d}$$

Since the LP relaxation and its dual are infinitely-sized, the strong duality property does not hold automatically, as in the finite LP case. However, we do not need strong duality for our approach. We rely only on the fact that the optimal value of integer program (9) is at least that of its LP relaxation, and on weak duality between the LP relaxation and its dual.

2.2 Analysis of a Single Facility

In this section, we prove several key lemmas that will enable us to execute Algorithm FLPD implicitly. We prove the lemmas in a simplified setting when problem (4) has only

one facility, and all connection costs c_{ij} are zero. 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 when the increase in the dual variables stops. The increase may stop because a closed tangent became tight, or because an unconnected customer began contributing to an open tangent. We assume that there are no open tangents, which implies that the increase stops because a closed tangent became tight.

Let $t = 0$ at the beginning of the step, and imagine that t is increasing to $+\infty$. The customer budgets start at $v_i \geq 0$ and increase over time at rates δ_i. At time t, the budget for customer i has increased to $v_i + t\delta_i$. Connected customers are modeled by taking $\delta_i = 0$, and unconnected customers by taking $\delta_i = d_i$. Denote the set of connected customers by C and the set of unconnected customers by U, and let $\mu = |U|$.

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

Lemma 1. *If $v_i = 0$ for $i \in [m]$, then tangent $p^* = \sum_{i \in U} d_i$ becomes tight first, at time $t^* = s_{p^*} + \frac{f_{p^*}}{p^*}$. If there is a tie, it is between at most two tangents.*

Proof. A given tangent p becomes tight at time $s_p + \frac{f_p}{\sum_{i \in U} d_i}$. Therefore,

$$p^* = \operatorname*{argmin}_{p \geq 0} \left\{ s_p + \frac{f_p}{\sum_{i \in U} d_i} \right\} = \operatorname*{argmin}_{p \geq 0} \left\{ s_p \sum_{i \in U} d_i + f_p \right\}. \tag{11}$$

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 ϕ, and ϕ is concave,

$$f_p + s_p \sum_{i \in U} d_i \geq \phi \left(\sum_{i \in U} d_i \right) \quad \text{for} \quad p \geq 0. \tag{12}$$

On the other hand, for tangent $p^* = \sum_{i \in U} d_i$, we have $f_{p^*} + s_{p^*} \sum_{i \in U} d_i = \phi\big(\sum_{i \in U} d_i\big)$. Therefore, tangent p^* becomes tight first, at time $t^* = s_{p^*} + \frac{f_{p^*}}{p^*}$. (See Figure 1.)

Concerning ties, for a tangent p to become tight first, it has to satisfy $f_p + s_p \sum_{i \in U} d_i = \phi\big(\sum_{i \in U} d_i\big)$, or in other words it has to be tangent to ϕ at $\sum_{i \in U} d_i$. We consider two cases. First, let ζ_2 be as large as possible so that ϕ is linear on $[p^*, \zeta_2]$. Then, any point $p \in [p^*, \zeta_2)$ yields the same tangent as p^*, that is $(f_p, s_p) = (f_{p^*}, s_{p^*})$. Second, let ζ_1 be as small as possible so that ϕ is linear on $[\zeta_1, p^*]$. Then, any point $p \in [\zeta_1, p^*)$ yields the same tangent as ζ_1, that is $(f_p, s_p) = (f_{\zeta_1}, s_{\zeta_1})$. Tangent ζ_1 is also tangent to ϕ at $\sum_{i \in U} d_i$, and may be different from tangent p^*. Tangents $p \notin [\zeta_1, \zeta_2]$ have $f_p + s_p \sum_{i \in U} d_i > \phi\big(\sum_{i \in U} d_i\big)$. Therefore, in a tie, at most two tangents, ζ_1 and p^*, become tight first. □

Next, we return to the more general case when customers have nonnegative starting budgets. Define

$$p_i(t) = \min\{p \geq 0 : v_i + t\delta_i \geq s_p d_i\}, \qquad i \in [m], \tag{13}$$

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

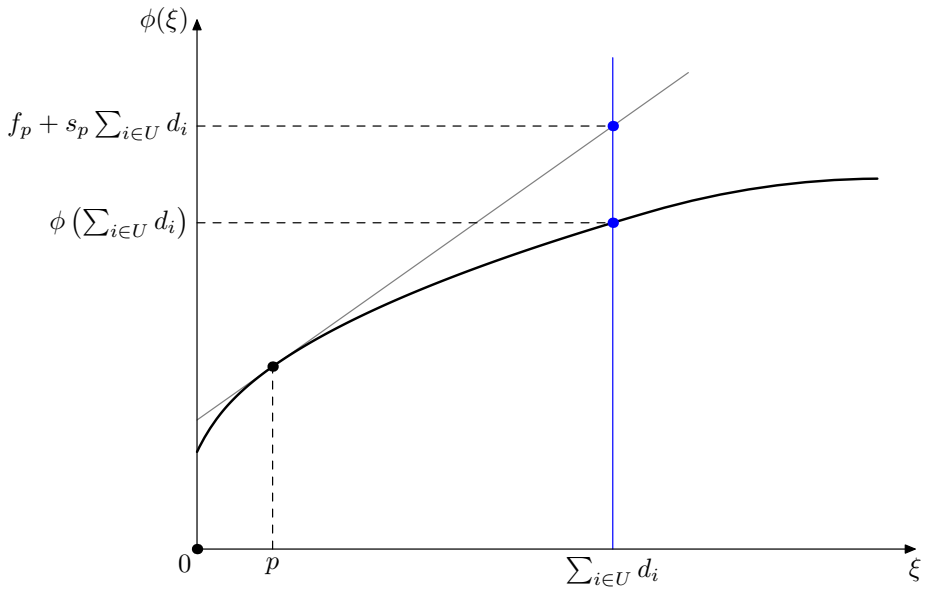

Figure 1: Illustration of the proof of Lemma 1.

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,

$$(v_i + t\delta_i)/d_i > (v_j + t\delta_j)/d_j \quad \Rightarrow \quad p_i(t) \le p_j(t), \tag{14a}$$
$$(v_i + t\delta_i)/d_i = (v_j + t\delta_j)/d_j \quad \Rightarrow \quad p_i(t) = p_j(t), \tag{14b}$$
$$(v_i + t\delta_i)/d_i < (v_j + t\delta_j)/d_j \quad \Rightarrow \quad p_i(t) \ge p_j(t). \tag{14c}$$

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

$$v_1/d_1 \ge v_2/d_2 \ge \cdots \ge v_\mu/d_\mu, \tag{15a}$$
$$v_{\mu+1}/d_{\mu+1} \ge v_{\mu+2}/d_{\mu+2} \ge \cdots \ge v_m/d_m. \tag{15b}$$

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 (14), at all times t, we have $p_1(t) \le p_2(t) \le \cdots \le p_\mu(t)$ and $p_{\mu+1}(t) \le p_{\mu+2}(t) \le \cdots \le p_m(t)$. As t increases, $p_i(t)$ for $i \in C$ are unchanged, while $p_i(t)$ for $i \in U$ decrease. (See Figure 2.)

Let

$$I_k^{\mathrm{u}}(t) = [p_k(t), p_{k+1}(t)), \qquad 1 \le k < \mu, \tag{16a}$$
$$I_l^{\mathrm{c}}(t) = [p_l(t), p_{l+1}(t)), \qquad \mu + 1 \le l < m, \tag{16b}$$

with $I_0^{\mathrm{u}}(t) = [0, p_1(t))$ and $I_\mu^{\mathrm{u}}(t) = [p_\mu(t), +\infty)$, as well as $I_\mu^{\mathrm{c}}(t) = [0, p_{\mu+1}(t))$ and $I_m^{\mathrm{c}}(t) = [p_m(t), +\infty)$. When an interval has the form $[+\infty, +\infty)$, we interpret it to be empty.

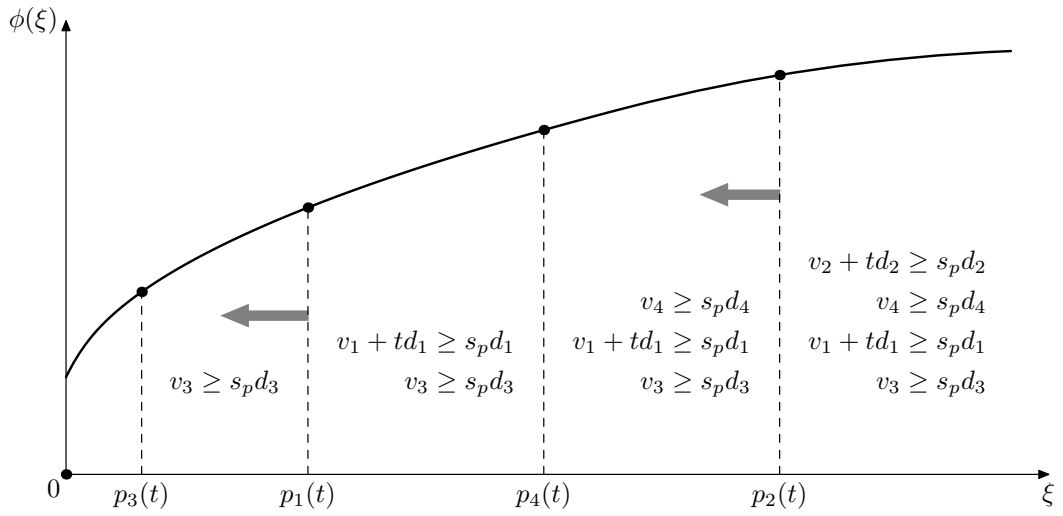

Figure 2: Illustration of the definition of $p_i(t)$. Here $U = \{1, 2\}$ and $C = \{3, 4\}$. The gray arrows show how $p_i(t)$ change as t increases. The inequalities show the set of customers that contribute to the tangents in each of the intervals defined by $p_i(t)$.

Consider the intervals

$$I_{kl}(t) = I_k^{\mathrm{u}}(t) \cap I_l^{\mathrm{c}}(t), \qquad 0 \leq k \leq \mu \leq l \leq m. \tag{17}$$

At any given time t, some of the intervals $I_{kl}(t)$ may be empty. As time increases, these intervals may vary in size, empty intervals may become non-empty, and non-empty intervals may become empty. The intervals partition $[0, +\infty)$, that is $\cup_{0 \leq k \leq \mu \leq l \leq m} I_{kl}(t) = [0, +\infty)$, and $I_{kl}(t) \cap I_{rs}(t) = \emptyset$ for $(k, l) \neq (r, s)$.

Let $\omega_p(t)$ be the total contribution received by tangent p at time t. The tangents on each interval $I_{kl}(t)$ receive contributions from unconnected customers $\{1, \ldots, k\}$ and connected customers $\{\mu + 1, \ldots, l\}$. We define $C(k, l) = \{1, \ldots, k\} \cup \{\mu + 1, \ldots, l\}$ to be the set of customers that contribute to tangents in $I_{kl}(t)$.

For each interval $I_{kl}(t)$ with $k \geq 1$, we define an alternate setting $A(k, l)$, where all starting budgets are zero, customers in $C(k, l)$ increase their budgets at rates d_i, and the remaining customers do not change their budgets. Let $\omega_p^{kl}(\tau_{kl})$ be the total contribution received by tangent p at time τ_{kl} in the alternate setting $A(k, l)$. We establish a correspondence between times t in the original setting and times τ_{kl} in $A(k, l)$, given by $\tau_{kl} = \beta_{kl} + \alpha_{kl} t$, with $\alpha_{kl} = \sum_{i=1}^{k} d_i \big/ \sum_{i \in C(k,l)} d_i$ and $\beta_{kl} = \sum_{i \in C(k,l)} v_i \big/ \sum_{i \in C(k,l)} d_i$. Since $\alpha_{kl} > 0$, times $t \in [0, +\infty)$ are mapped one-to-one to times $\tau_{kl} \in [\beta_{kl}, +\infty)$.

The following two lemmas relate the original setting to the alternate settings $A(k, l)$.

Lemma 2. *Given a time t and an interval $I_{kl}(t)$ with $k \geq 1$, any tangent $p \in I_{kl}(t)$ receives the same total contribution at time t in the original setting as at time τ_{kl} in $A(k, l)$, that is $\omega_p(t) = \omega_p^{kl}(\tau_{kl})$.*

Proof. The total contribution to p at time t in the original setting is

$$\begin{aligned}\omega_p(t) &= \sum_{i=1}^{k}(v_i + td_i - s_p d_i) + \sum_{i=\mu+1}^{l}(v_i - s_p d_i) \\ &= \sum_{i\in C(k,l)}(v_i - s_p d_i) + t\sum_{i=1}^{k} d_i = \sum_{i\in C(k,l)}(v_i + \alpha_{kl} t d_i - s_p d_i) \\ &= (\beta_{kl} + \alpha_{kl} t - s_p)\sum_{i\in C(k,l)} d_i = (\tau_{kl} - s_p)\sum_{i\in C(k,l)} d_i.\end{aligned} \tag{18}$$

Since $\omega_p(t) \geq 0$, it follows that $\tau_{kl} - s_p \geq 0$, and therefore

$$(\tau_{kl} - s_p)\sum_{i\in C(k,l)} d_i = \sum_{i\in C(k,l)} \max\{0, \tau_{kl} d_i - s_p d_i\} = \omega_p^{kl}(\tau_{kl}). \tag{19}$$

□

Lemma 3. *Given a time t and an interval $I_{kl}(t)$ with $k \geq 1$, a tangent p receives at least as large a total contribution at time t in the original setting as at time τ_{kl} in $A(k,l)$, that is $\omega_p(t) \geq \omega_p^{kl}(\tau_{kl})$.*

Proof. If $\tau_{kl} - s_p < 0$, then $\omega_p^{kl}(\tau_{kl}) = 0$, and thus $\omega_p(t) \geq \omega_p^{kl}(\tau_{kl})$. If $\tau_{kl} - s_p \geq 0$, then $\omega_p^{kl}(\tau_{kl}) = (\tau_{kl} - s_p)\sum_{i\in C(k,l)} d_i = \sum_{i=1}^{k}(v_i + td_i - s_p d_i) + \sum_{i=\mu+1}^{l}(v_i - s_p d_i)$. Let $p \in I_{rs}(t)$ for some r and s, and note that $\omega_p(t) = \sum_{i=1}^{r}(v_i + td_i - s_p d_i) + \sum_{i=\mu+1}^{s}(v_i - s_p d_i)$.

The difference between the two contributions can be written as

$$\begin{aligned}\omega_p(t) - \omega_p^{kl}(\tau_{kl}) = &\sum_{i=k+1}^{r}(v_i + td_i - s_p d_i) - \sum_{i=r+1}^{k}(v_i + td_i - s_p d_i) \\ &+ \sum_{i=l+1}^{s}(v_i - s_p d_i) - \sum_{i=s+1}^{l}(v_i - s_p d_i).\end{aligned} \tag{20}$$

Note that at least two of the four summations in this expression are always empty. We now examine the summations one by one:

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

As a result of the above cases, we obtain that $\omega_p(t) - \omega_p^{kl}(\tau_{kl}) \geq 0$. □

When $k \geq 1$, we can apply Lemma 1 to compute the first tangent to become tight in $A(k,l)$, and the time when this occurs. Denote the computed tangent and time by p'_{kl} and τ'_{kl}, and note that $p'_{kl} = \sum_{i \in C(k,l)} d_i$ 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 $A(k,l)$. Let $t^* = \min\{t'_{kl} : 1 \leq k \leq \mu \leq l \leq m\}$, and $p^* = p'_{\text{argmin}\{t'_{kl}:1\leq k\leq\mu\leq l\leq m\}}$. The following two lemmas will enable us to show that tangent p^* becomes tight first in the original setting, at time t^*.

Lemma 4. *If a tangent p becomes tight at a time t in the original setting, then $t \geq t^*$.*

Proof. Since $\cup_{0\leq r\leq\mu\leq s\leq m} I_{rs}(t) = [0, +\infty)$, there is an interval $I_{kl}(t)$ that contains p. Since the contributions to tangents in the interval $I_{0l}(t)$ do not increase over time, $k \geq 1$.

Tangent p is tight at time t in the original setting, and therefore $\omega_p(t) = f_p$. By Lemma 2, $p \in I_{kl}(t)$ implies that $\omega_p^{kl}(\tau_{kl}) = f_p$, and hence p is tight at time τ_{kl} in $A(k,l)$. It follows that $\tau_{kl} \geq \tau'_{kl}$, and therefore $t \geq t'_{kl} \geq t^*$. □

Lemma 5. *Each tangent p'_{kl} with $k \geq 1$ becomes tight at a time $t \leq t'_{kl}$ in the original setting.*

Proof. Since tangent p'_{kl} is tight at time τ'_{kl} in $A(k,l)$, we have $\omega_{p'_{kl}}^{kl}(\tau'_{kl}) = f_{p'_{kl}}$. By Lemma 3, $\omega_{p'_{kl}}(t'_{kl}) \geq f_{p'_{kl}}$, which means that p'_{kl} is tight or over-tight at time t'_{kl} in the original setting. Therefore, p'_{kl} becomes tight at a time $t \leq t'_{kl}$ in the original setting. □

We now obtain the main result of this section.

Lemma 6. *Tangent p^* becomes tight first in the original setting, at time t^*. The quantities p^* and t^* can be computed in time $O(m^2)$.*

Proof. Lemma 4 implies that tangents become tight only at times $t \geq t^*$. Lemma 5 implies that tangent $p^* = p'_{\text{argmin}\{t'_{kl}:1\leq k\leq\mu\leq l\leq m\}}$ becomes tight at a time $t \leq \min\{t'_{kl} : 1 \leq k \leq \mu \leq l \leq m\} = t^*$. Therefore, tangent p^* becomes tight first, at time t^*.

To evaluate the running time, note that the d_i and v_i can be sorted in $O(m \log m)$ time. Once the d_i and v_i are sorted, we can compute all quantities α_{kl} and β_{kl} in $O(m^2)$, and then compute all t'_{kl} and p'_{kl} in $O(m^2)$ via Lemma 1. Therefore, the total running time is $O(m^2)$. □

In case of a tie, Lemma 6 enables us to compute one of the tangents that become tight first. It is possible to obtain additional results about ties, starting with that of Lemma 1. However, we do not need such results in this paper, as Algorithm FLPD, as well as the algorithms in Sections 3 and 4, allow us to break ties arbitrarily.

For many primal-dual algorithms, we can perform the computation in Lemma 6 faster than in $O(m^2)$, by taking into account the details of how the algorithm increases the dual variables. We will illustrate this with three algorithms in Sections 2.4, 3, and 4.

2.3 Other Rules for Changing the Dual Variables

In this section, we consider the same setting as in the previous one, but in addition allow each customer i to change its budget at an arbitrary rate $\delta_i \geq 0$. The rate is no longer limited to the set $\{0, d_i\}$, and we assume that at least one customer has $\delta_i > 0$. The following results are not needed to obtain the algorithms in this paper. We include them since they embody a more general version of our approach, and may be useful in developing primal-dual algorithms in the future.

Consider the quantities $p_i(t)$ as defined in equation (13). Since δ_i need not equal d_i, the order of the $p_i(t)$ may change as t increases from 0 to $+\infty$. At any given time t, the $p_i(t)$ divide $[0, +\infty)$ into at most $m + 1$ intervals. For each set of customers $K \subseteq [m]$, we introduce an interval

$$I_K(t) = [a_K(t), b_K(t)) = \Big[\max_{i \in K} p_i(t), \min_{i \notin K} p_i(t)\Big). \tag{21}$$

If $K = \emptyset$, we set $a_K(t) = 0$, and if $K = [m]$, we set $b_K(t) = +\infty$. If $a_K(t) \geq b_K(t)$ or $a_K(t) = b_K(t) = +\infty$, we take $I_K(t)$ to be empty. Note that $\bigcup_{K \subseteq [m]} I_K(t) = [0, +\infty)$, and $I_K(t) \cap I_L(t) = \emptyset$ for $K \neq L$. Any interval that is formed by the $p_i(t)$ as t increases from 0 to $+\infty$ is among the intervals $I_K(t)$. The set of customers contributing to tangents on an interval $I_K(t)$ is precisely K.

As in the previous section, for each interval $I_K(t)$ with $\sum_{i \in K} \delta_i > 0$, we define an alternate setting $A(K)$, where all starting budgets are zero, customers in K increase their budgets at rates d_i, and the remaining customers keep their budgets unchanged. We denote the total contribution received by tangent p at time τ_K in $A(K)$ by $\omega_p^K(\tau_K)$. The correspondence between times t in the original setting and times τ_K in $A(K)$ is given by $\tau_K = \beta_K + \alpha_K t$, with $\alpha_K = \sum_{i \in K} \delta_i / \sum_{i \in K} d_i$ and $\beta_K = \sum_{i \in K} v_i / \sum_{i \in K} d_i$. Since $\alpha_K > 0$, the correspondence is one-to-one between times $t \in [0, +\infty)$ and $\tau_K \in [\beta_K, +\infty)$.

Lemma 7. *Given a time t and an interval $I_K(t)$ with $\sum_{i \in K} \delta_i > 0$, a tangent $p \in I_K(t)$ receives the same total contribution at time t in the original setting as at time τ_K in $A(K)$, that is $\omega_p(t) = \omega_p^K(\tau_K)$.*

Proof. The total contribution in the original setting is

$$\begin{aligned} \omega_p(t) &= \sum_{i \in K} (v_i + t\delta_i - s_p d_i) = \sum_{i \in K} (v_i + \alpha_K t d_i - s_p d_i) \\ &= (\beta_K + \alpha_K t - s_p) \sum_{i \in K} d_i = (\tau_K - s_p) \sum_{i \in K} d_i = \omega_p^K(\tau_K). \end{aligned} \tag{22}$$

□

Lemma 8. *Given a time t and an interval $I_K(t)$ with $\sum_{i \in K} \delta_i > 0$, a tangent p receives at least as large a total contribution at time t in the original setting as at time τ_K in $A(K)$, that is $\omega_p(t) \geq \omega_p^K(\tau_K)$.*

Proof. If $\tau_K - s_p < 0$, then $\omega_p(t) \geq \omega_p^K(\tau_K)$. If $\tau_K - s_p \geq 0$, let $p \in I_L(t)$ for some $L \subseteq [m]$, and note that $\omega_p^K(\tau_K) = (\tau_K - s_p) \sum_{i \in K} d_i = \sum_{i \in K} (v_i + t\delta_i - s_p d_i)$, while $\omega_p(t) = \sum_{i \in L} (v_i + t\delta_i - s_p d_i)$.

The difference between the two contributions is

$$\omega_p(t) - \omega_p^K(\tau_K) = \sum_{i \in L \setminus K} (v_i + t\delta_i - s_p d_i) - \sum_{i \in K \setminus L} (v_i + t\delta_i - s_p d_i). \tag{23}$$

Since $p \in I_L(t)$, in the original setting, customers in L contribute to tangent p at time t, and therefore $v_i + t\delta_i - s_p d_i \geq 0$ for $i \in L$, which implies that $\sum_{i \in L \setminus K}(v_i + t\delta_i - s_p d_i) \geq 0$. Conversely, customers not in L do not contribute to p at time t, implying that $v_i + t\delta_i - s_p d_i \leq 0$ for $i \notin L$, and therefore $\sum_{i \in K \setminus L}(v_i + t\delta_i s_p d_i) \leq 0$. As a result, $\omega_p(t) - \omega_p^K(\tau_K) \geq 0$. □

Unlike in the previous section, we have exponentially many alternative settings $A(K)$. The following derivations will enable us to compute the first tangent to become tight in the original setting, and the time when this occurs using only a polynomial number of alternative settings.

As t increases from 0 to $+\infty$, the order of the quantities $(v_i + t\delta_i)/d_i$ may change. Since the quantities are linear in t, as $t \to +\infty$, they assume an order that no longer changes. We use this order to define a permutation $\pi(+\infty) = (\pi_1(+\infty), \ldots, \pi_m(+\infty))$, with $\pi_i(+\infty) = j$ meaning that $(v_j + t\delta_j)d_j$ is the i-th largest quantity. If two quantities are tied as $t \to +\infty$, we break the tie arbitrarily. Similarly, for any time $t \in [0, +\infty)$, we define a permutation $\pi(t) = (\pi_1(t), \ldots, \pi_m(t))$. In this case, if two quantities are tied, we break the tie according to $\pi(+\infty)$. For example, suppose that the two largest quantities at time t are tied, that they are $(v_1 + t\delta_1)/d_1 = (v_2 + t\delta_2)/d_2$, and that $\pi_i(+\infty) = 1$ and $\pi_j(+\infty) = 2$ with $i < j$. Then we take $\pi_1(t) = 1$ and $\pi_2(t) = 2$.

Compare two quantities

$$(v_i + t\delta_i)/d_i \quad \text{vs.} \quad (v_j + t\delta_j)/d_j. \tag{24}$$

If their order changes as t increases from 0 to $+\infty$, then there is a $\theta > 0$ such that the sign between the quantities is '$<$' on $[0, \theta)$, '$=$' at θ, and '$>$' on $(\theta, +\infty)$, or vice-versa. Let $\theta_1 < \cdots < \theta_R$ be all such times when the sign between two quantities changes, and let $\theta_0 = 0$ and $\theta_{R+1} = +\infty$. Since there are $m(m-1)/2$ pairs of quantities, $R \leq m(m-1)/2$.

The proof of the following lemma follows from these definitions.

Lemma 9. *As t increases from 0 to $+\infty$, the permutation $\pi(t)$ changes at times $\theta_1, \ldots, \theta_R$. Moreover, $\pi(t)$ is unchanged on the intervals $[\theta_r, \theta_{r+1})$ for $r = 0, \ldots, R$.*

We now bound the number of intervals $I_K(t)$ that ever become nonempty. Let $\mathcal{K}(t) = \{\{\pi_1(t), \ldots, \pi_i(t)\} : i = 0, \ldots, m\}$ and $\mathcal{K} = \cup_{r=0}^{R} \mathcal{K}(\theta_r)$, and note that $|\mathcal{K}(t)| \leq m + 1$ and $|\mathcal{K}| \leq (m+1)(m(m-1)/2 + 1) = O(m^3)$.

Lemma 10. *As t increases from 0 to $+\infty$, only intervals $I_K(t)$ with $K \in \mathcal{K}$ ever become nonempty, that is $\{K : \exists t \geq 0 \text{ s.t. } I_K(t) \neq \emptyset\} \subseteq \mathcal{K}$.*

Proof. Fix a time t, and note that by property (14), we have $p_{\pi_1(t)}(t) \leq p_{\pi_2(t)}(t) \leq \cdots \leq p_{\pi_m(t)}(t)$. Therefore, the intervals $I_K(t)$ may be nonempty only when $K \in \mathcal{K}(t)$. Since $\pi(t)$ is unchanged on the intervals $[\theta_r, \theta_{r+1})$ for $r = 0, \ldots, R$, if an interval $I_K(t)$ ever becomes nonempty, then $K \in \mathcal{K}$. □

As in the previous section, when $\sum_{i \in K} \delta_i > 0$, we can compute the first tangent to become tight in $A(K)$, and the time when this occurs using Lemma 1. Let the computed tangent and time be $p'_K = \sum_{i \in K} d_i$ and $\tau'_K = s_{p'_K} + \frac{f_{p'_K}}{p'_K}$, and let $t'_K = \frac{\tau'_K - \beta_K}{\alpha_K}$ be the time in the original setting corresponding to time τ'_K in $A(K)$. Next, we show that tangent $p^* = p'_{\operatorname{argmin}\{t'_K : \sum_{i \in K} \delta_i > 0, K \in \mathcal{K}\}}$ becomes tight first, at time $t^* = \min\{t'_K : \sum_{i \in K} \delta_i > 0, K \in \mathcal{K}\}$.

Lemma 11. *If a tangent p becomes tight at a time t in the original setting, then $t \geq t^*$.*

Proof. Let $I_K(t)$ be the interval that contains p. Since this interval is nonempty, $K \in \mathcal{K}$, and since the contribution to p must be increasing over time, $\sum_{i \in K} \delta_i > 0$.

Tangent p is tight at time t in the original setting, and therefore $\omega_p(t) = f_p$. By Lemma 7, $\omega_p^K(\tau_K) = f_p$, and hence p is tight at time τ_K in $A(K)$. It follows that $\tau_K \geq \tau'_K$, and therefore $t \geq t'_K \geq t^*$. □

Lemma 12. *Each tangent p'_K with $\sum_{i \in K} \delta_i > 0$ becomes tight at a time $t \leq t'_K$ in the original setting.*

Proof. Since p'_K is tight at time τ'_K in $A(K)$, we have $\omega_{p'_K}^K(\tau'_K) = f_{p'_K}$. By Lemma 8, $\omega_{p'_K}(t'_K) \geq f_{p'_K}$, which means that p'_K is tight or over-tight at time t'_K in the original setting. Therefore, p'_K becomes tight at a time $t \leq t'_K$ in the original setting. □

Lemma 13. *Tangent p^* becomes tight first in the original setting, at time t^*. The quantities p^* and t^* can be computed in time $O(m^3)$.*

Proof. By Lemma 11, tangents only become tight at times $t \geq t^*$, while by Lemma 12, $p^* = p'_{\operatorname{argmin}\{t'_K : \sum_{i \in K} \delta_i > 0, K \in \mathcal{K}\}}$ becomes tight at a time $t \leq \min\{t'_K : \sum_{i \in K} \delta_i > 0, K \in \mathcal{K}\} = t^*$. Therefore, p^* becomes tight first, at time t^*.

Concerning the running time, note that the θ_r can be computed in $O(m^2)$ and sorted in $O(m^2 \log m)$ time. The permutations $\pi(+\infty)$ and $\pi(0)$ can be computed in $O(m \log m)$ time. Processing the θ_r in increasing order, we can compute each $\pi(\theta_r)$ in $O(m)$ time amortized over all θ_r. Computing p'_K and t'_K for all $K \in \mathcal{K}(\theta_r)$ takes $O(m)$ time. Therefore, the total running time is $O(m^3)$. □

The results in this section can be generalized further to allow the rates δ_i to be negative.

2.4 Analysis of Multiple Facilities

We now show how to execute Algorithm FLPD implicitly when problem (4) has multiple facilities. In Section 2.2, in addition to assuming the presence of only one facility, we assumed that the connection costs c_{ij} were 0. We remove this assumptions as well.

In this section, we continue to refer to facilities of infinitely-sized problem (9) as tangents, and reserve the term facility for facilities of concave cost problem (4). We say that customer i contributes to concave cost facility j if $v_i \geq c_{ij}$. We distinguish between when a customer contributes to a concave cost facility j and when the customer contributes to a tangent p belonging to concave cost facility j.

When executing Algorithm FLPD implicitly, the input consists of m, n, the connection costs c_{ij}, the demands d_i, and the cost functions ϕ_j, given by an oracle. As intermediate variables, we maintain the time t, and the vectors v, x, and y. For x and y, we maintain only the non-zero entries. The algorithm returns v, x, and y. We also maintain standard data structures to manipulate these quantities as necessary. Note that we do not maintain nor return the vector w, as any one of its entries can be computed through the invariant $w_{ijp} = \max\{0, v_i - (c_{ij} + s_{jp})d_i\}$.

Clearly, step (1) can be executed in polynomial time. In order to use induction, suppose that we have executed at most $m-1$ iterations of loop (2) so far. Since the algorithm opens at most one tangent at each iteration, at any point at most $m-1$ tangents are open. To analyze step (3), we consider three events that may occur as this step is executed:

1. A closed tangent becomes tight.
2. An unconnected customer begins contributing to an open tangent.
3. An unconnected customer begins contributing to a facility.

When step (3) is executed, the time t stops increasing when event 1 or 2 occurs. For the purpose of analyzing this step, we assume that t increases to $+\infty$ and that v_i for unconnected customers are increased so as to maintain $v_i = td_i$.

Lemma 14. *Suppose that event e at facility j is the first to occur after the beginning of step* (3)*. Then we can compute the time t' when this event occurs in polynomial time.*

Proof. If $e = 1$, we use Lemma 6 to compute t'. The lemma's assumptions can be satisfied as follows. Since no events occur at other facilities until time t', we can assume that j is the only facility. Since the set of customers contributing to facility j will not change until time t', we can satisfy the assumption that $c_{ij} = 0$ by subtracting c_{ij} from each v_i having $v_i \geq c_{ij}$. Since an unconnected customer will not begin contributing to an open tangent until time t', we can assume that there are no open tangents. We can satisfy the assumption that $t = 0$ at the beginning of step (3) by adding td_i to each v_i.

If $e = 2$, we compute t' by iterating over all unconnected customers and open tangents of facility j. If $e = 3$, we compute t' by iterating over all unconnected customers. □

When other events occur between the beginning of step (3) and time t', the computation in this lemma may be incorrect, however we can still perform it. Let $t'_e(j)$ be the time computed in this manner for a given e and j, and let $t^* = \min\{t'_e(j) : e \in [3], j \in [n]\}$ and $(e^*, j^*) = \operatorname{argmin}\{t'_e(j) : e \in [3], j \in [n]\}$.

Lemma 15. *Event e^* at facility j^* is the first to occur after the beginning of step* (3)*. This event occurs at time t^*.*

Proof. Suppose that an event e' at a facility j' occurs at a time $t' < t^*$. If $e' \in \{2, 3\}$, then $t' \geq t'_{e'}(j') \geq t^*$. This is a contradiction, and therefore this case cannot occur.

If $e' = 1$, then we consider two cases. If there is an event $e'' \in \{2, 3\}$ that occurs at a facility j'' at a time $t'' < t'$, then we use $t'' \geq t'_{e''}(j'') \geq t^*$ to obtain a contradiction. If there is no such event e'', then no new customer begins contributing to facility j' between the beginning of step (3) and time t'. Therefore, $t' \geq t'_{e'}(j') \geq t^*$, and we again obtain a contradiction. □

Once we have computed t^*, e^*, and j^*, we finish executing step (3) as follows. If $e^* = 3$, that is if the first event to occur is an unconnected customer beginning to contribute to j^*, we update the list of customers contributing to j^* and recompute t^*, e^*, and j^*. Since there are n facilities and at most m unconnected customers, event 3 can occur at most mn times before event 1 or 2 takes place.

Once event 1 or 2 takes place, step (3) is complete, and we have to execute step (4) or (5). It is easy to see that these steps can be executed in polynomial time. Therefore, an additional iteration of loop (2) can be executed in polynomial time. By induction, each of the first m iterations of loop (2) can be executed in polynomial time. At each iteration, an unconnected customer is connected, either in step (4) or (5). Therefore, loop (2) iterates at most m times. Obviously, step (6) can be executed in polynomial time, and therefore Algorithm FLPD can be executed implicitly in polynomial time. Recall that we called the algorithm obtained by executing FLPD implicitly on infinitely-sized problem (9) ConcaveFLPD.

Theorem 2. *Algorithm* ConcaveFLPD *is a 1.61-approximation algorithm for concave cost facility location, with a running time of* $O(m^3n + mn \log n)$.

Proof. At the beginning of the algorithm, we sort the connection costs c_{ij}, which can be done in $O(mn \log(mn))$ time. Next, we bound the time needed for one iteration of loop (2). Note that since loop (2) iterates at most m times, there are at most m open tangents at any point in the algorithm.

In step (3), we first compute $\min\{t'_1(j) : j \in [n]\}$. Computing each $t'_1(j)$ requires $O(m^2)$ per facility, and thus this part takes $O(m^2n)$ overall. Next, we compute $\min\{t'_2(j) : j \in [n]\}$, using $O(1)$ per customer and open tangent, and thus $O(m^2)$ overall. Finally, we compute $\min\{t'_3(j) : j \in [n]\}$. Since $\min\{t'_3(j) : j \in [n]\} = \min\{c_{ij} : c_{ij} \geq t\}$, we have sorted the values c_{ij}, and t only increases as the algorithm runs, this operation takes $O(mn)$ over the entire run of the algorithm. Therefore, we can determine the next event to occur in $O(m^2n)$.

If event 1 or 2 is the next one, step (3) is complete. If event 3 is next, an unconnected customer begins to contribute to facility j^*. In this case, we recompute $t'_1(j^*)$ and $\min\{t'_1(j) : j \in [n]\}$. Recomputing $t'_1(j^*)$ can be done in $O(m)$, since we only have to add one customer to the setting of Lemma 6. Recomputing $\min\{t'_1(j) : j \in [n]\}$ takes $O(1)$, as $t'_1(j^*)$ does not increase when an unconnected customer begins contributing to facility j^*. Note that $\min\{t'_2(j) : j \in [n]\}$ does not change. Next, we recompute $\min\{t'_3(j) : j \in [n]\}$, which takes $O(mn)$ over the entire run of the algorithm. The total time to process event 3 and determine the next event to occur is $O(m)$. Event 3 occurs at most mn times before event 1 or 2 occurs, and therefore the total time for processing event 3 occurrences is $O(m^2n)$.

Step (4) can be done in $O(m)$, and step (5) in $O(1)$. Therefore, the time for one iteration of loop (2) is $O(m^2n)$. Since there are at most m iterations of loop (2), the running time of the algorithm is $O(m^3n + mn \log n)$.

By Theorem 1, Algorithm FLPD is a 1.61-approximation algorithm for problem (1). The approximation ratio for problem (4) follows directly from the fact that we execute Algorithm FLPD implicitly on infinitely-sized problem (9). □

By a similar application of our technique to the 1.861-approximation algorithm for classical facility location 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 Concave Cost Lot-Sizing

In this section, we apply the technique developed in Section 2 to concave cost lot-sizing. The classical lot-sizing problem is defined in Section 1.1.2, and can be written as a linear program:

$$\min \sum_{s=1}^{n} f_s y_s + \sum_{s=1}^{n} \sum_{t=s}^{n} (c_s + h_{st}) d_t x_{st}, \tag{25a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st} = 1, \qquad 1 \le t \le n, \tag{25b}$$

$$0 \le x_{st} \le y_s, \qquad 1 \le s \le t \le n. \tag{25c}$$

Recall that $f_t \in \mathbb{R}_+$ and $c_t \in \mathbb{R}_+$ are the fixed and per-unit costs of placing an order at time t, and $d_t \in \mathbb{R}_+$ is the demand at time t. The per-unit holding cost at time t is $h_t \in \mathbb{R}_+$, and for convenience, we defined $h_{st} = \sum_{i=s}^{t-1} h_i$. Note that we omit the constraints $y_s \in \{0, 1\}$, as there is always an optimal extreme point solution that satisfies them [KB77].

We now adapt the algorithm of Levi et al. [LRS06] to work in the setting of problem (25). Levi et al. derive their algorithm in a slightly different setting, where the costs h_{st} are not necessarily the sum of period holding costs h_t, but rather satisfy an additional monotonicity condition.

The dual of problem (25) is given by:

$$\max \sum_{t=1}^{n} v_t, \tag{26a}$$

$$\text{s.t. } v_t \le (c_s + h_{st}) d_t + w_{st}, \qquad 1 \le s \le t \le n, \tag{26b}$$

$$\sum_{t=s}^{n} w_{st} \le f_s, \qquad 1 \le s \le n, \tag{26c}$$

$$w_{st} \ge 0, \qquad 1 \le s \le t \le n. \tag{26d}$$

As with facility location, since the variables w_{st} do not appear in the objective, we assume 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 $v_t \ge (c_s + h_{st}) d_t$, we say that demand point t *contributes* to order s, and w_{st} is its contribution. The total contribution received by an order s is $\sum_{t=s}^{n} w_{st}$. An order t is *tight* if $\sum_{t=s}^{n} w_{st} = f_s$ and *over-tight* if $\sum_{t=s}^{n} w_{st} > f_s$.

The primal complementary slackness constraints are:

$$x_{st}(v_t - (c_s + h_{st})d_t - w_{st}) = 0, \qquad 1 \le s \le t \le n, \tag{27a}$$

$$y_s \left(\sum_{t=s}^{n} w_{st} - f_s \right) = 0, \qquad 1 \le s \le n. \tag{27b}$$

Let (x, y) be an integral primal feasible solution, and (v, w) be a dual feasible solution. Constraint (27a) 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 (27b) 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., as adapted here, 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 primal 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. In the algorithm, 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).

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

Theorem 3. *Algorithm* LSPD *is an exact algorithm for the classical lot-sizing problem.*

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. □

3.1 Applying the Technique

We now proceed to develop an exact primal-dual algorithm for concave cost lot-sizing. The concave cost lot-sizing problem is defined in Section 1.1.2:

$$\min \sum_{s=1}^{n} \phi_s \left(\sum_{t=s}^{n} d_t x_{st} \right) + \sum_{s=1}^{n} \sum_{t=s}^{n} h_{st} d_t x_{st}, \tag{28a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st} = 1, \qquad 1 \le t \le n, \tag{28b}$$

$$x_{st} \ge 0, \qquad 1 \le s \le t \le n. \tag{28c}$$

Here, the cost of placing an order at time t is given by a nondecreasing concave cost function $\phi_t : \mathbb{R}_+ \to \mathbb{R}_+$. We assume without loss of generality that $\phi_t(0) = 0$ for all t.

The application of our technique to the lot-sizing problem is similar to its application to the facility location problem in Section 2. First, we reduce concave cost lot-sizing problem

(28) to the following infinitely-sized classical lot-sizing problem.

$$\min \sum_{s=1}^{n} \sum_{p \geq 0} f_{sp} y_{sp} + \sum_{s=1}^{n} \sum_{t=s}^{n} \sum_{p \geq 0} (c_{sp} + h_{st}) d_t x_{spt}, \tag{29a}$$

$$\text{s.t.} \sum_{s=1}^{t} \sum_{p \geq 0} x_{spt} = 1, \qquad 1 \leq t \leq n, \tag{29b}$$

$$0 \leq x_{spt} \leq y_{sp}, \qquad 1 \leq s \leq t \leq n, p \geq 0. \tag{29c}$$

Again we note that since LP (29) is infinitely-sized, strongly duality does not hold automatically for it and its dual. However, the proof of Algorithm LSPD relies only on weak duality. The fact that the algorithm produces a primal solution and a dual solution with the same cost implies that both solutions are optimal and that strong duality holds.

Following Section 2, let ConcaveLSPD be the algorithm obtained by executing Algorithm LSPD implicitly on infinitely-sized problem (29).

Theorem 4. *Algorithm* ConcaveLSPD *is an exact algorithm for concave cost lot-sizing, with a running time of* $O(n^2)$.

Proof. We consider the following events that may occur as step (3) of Algorithm LSPD is executed:

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 let $W_2(t) = +\infty$. 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 $W_2'(t)$. Each value $W_2'(t)$ is defined as the time when a tangent of order point t becomes tight in a truncated problem consisting of time periods $t, t+1, \ldots, n$. We compute a subset of these values as follows. First, we compute $W_2'(n)$, which requires $O(1)$ time by Lemma 1. To compute $W_2'(t)$ given that $W_2'(t+1), \ldots, W_2'(n)$ are computed, we can employ Lemma 6.

The dual variables representing demand points $t, \ldots, n$ can be divided into three consecutive intervals. 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 play no role in this computation. We employ Lemma 6 and distinguish two cases:

1. Lemma 6 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. In this case, we delete $W_2'(t+1)$ from our subset 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 less 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 subset 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 subset yield the correct times $W_2(t)$, with the other values $W_2(t) = +\infty$. Therefore, loop (2) is complete.

A computation by Lemma 6 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 6 in $O(n)$. Each time we use Lemma 6 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 $W_2(\sigma(t)) < +\infty$.

4 Concave Cost Joint Replenishment

In this section, we apply our technique to the concave cost joint replenishment problem (JRP). The classical JRP is defined in Section 1.1.3, and can be formulated as an integer program:

$$\min \sum_{s=1}^{n} f^0 y_s^0 + \sum_{s=1}^{n}\sum_{k=1}^{K} f^k y_s^k + \sum_{s=1}^{n}\sum_{t=s}^{n}\sum_{k=1}^{K} h_{st}^k d_t^k x_{st}^k, \tag{30a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st}^k = 1, \qquad 1 \le t \le n, k \in [K], \tag{30b}$$

$$0 \le x_{st}^k \le y_s^0, \qquad 1 \le s \le t \le n, k \in [K], \tag{30c}$$

$$0 \le x_{st}^k \le y_s^k, \qquad 1 \le s \le t \le n, k \in [K], \tag{30d}$$

$$y_s^0 \in \{0,1\}, y_s^k \in \{0,1\}, \qquad 1 \le s \le n, k \in [K]. \tag{30e}$$

Recall that $f^0 \in \mathbb{R}_+$ is the fixed joint ordering cost, $f^k \in \mathbb{R}_+$ is the fixed individual ordering cost for item k, and $d_t^k \in \mathbb{R}_+$ is the demand for item k at time t. The per-unit holding cost for item k at time t is h_t^k; for convenience we defined $h_{st}^k = \sum_{i=s}^{t-1} h_i^k$.

The concave cost JRP, also defined in Section 1.1.3, can be written as a mathematical program as follows:

$$\min \sum_{s=1}^{n} \phi^0\left(\sum_{t=s}^{n}\sum_{k=1}^{K} d_t^k x_{st}^k\right) + \sum_{s=1}^{n}\sum_{k=1}^{K} \phi^k\left(\sum_{t=s}^{n} d_t^k x_{st}^k\right) + \sum_{s=1}^{n}\sum_{t=s}^{n}\sum_{k=1}^{K} h_{st}^k d_t^k x_{st}^k, \tag{31a}$$

$$\text{s.t.} \sum_{s=1}^{t} x_{st}^k = 1, \qquad 1 \le t \le n, k \in [K], \tag{31b}$$

$$x_{st}^k \ge 0, \qquad 1 \le s \le t \le n, k \in [K]. \tag{31c}$$

Here the individual ordering cost for item k at time t is given by a nondecreasing concave function $\phi^k : \mathbb{R}_+ \to \mathbb{R}_+$. We assume without loss of generality that $\phi^k(0) = 0$ for all k. The joint ordering cost at time t is given by the function $\phi^0 : \mathbb{R}_+ \to \mathbb{R}_+$. To reflect the fact that only the individual ordering costs are general concave, ϕ^0 has the form $\phi^0(0) = 0$ and $\phi^0(\xi) = f^0$ for $\xi > 0$.

Consider the case when the individual ordering cost functions ϕ^k are piecewise linear with P pieces:

$$\phi^k(\xi_t^k) = \begin{cases} \min\{f_p^k + c_p^k \xi_t^k : p \in [P]\}, & \xi_t^k > 0, \\ 0, & \xi_t^k = 0, \end{cases} \tag{32}$$

Unlike with concave cost facility location and concave cost lot-sizing, the piecewise-linear concave cost JRP does not reduce polynomially to the classical JRP. Since there are multiple items, different pieces of the individual ordering cost functions ϕ^k may be employed by different items k as part of the same order at time t. When each cost function consists of P pieces, we would need P^K time periods to represent each possible combination, thereby leading to an exponentially-sized IP formulation.

We could devise a polynomially-sized IP formulation for the piecewise-linear concave cost JRP, however such a formulation would have a different structure from the classical JRP, and would not enable us to apply our technique together with the primal-dual algorithm of Levi et al. [LRS06] for the classical JRP. Instead, we reduce the piecewise-linear concave cost JRP to the following exponentially-sized integer programming formulation, which we call the *generalized joint replenishment* problem. Let $\pi = (p_1, \ldots, p_K)$, and let $[P]^K = \{(p_1, \ldots, p_K) : p_i \in [P]\}$.

$$\min \sum_{\substack{s \in [n] \\ \pi \in [P]^K}} f^0 y_{s\pi}^0 + \sum_{\substack{s \in [n], k \in [K] \\ \pi \in [P]^K}} f_{p_k}^k y_{s\pi}^k + \sum_{\substack{1 \le s \le t \le n \\ k \in [K], \pi \in [P]^K}} (c_{p_k}^k + h_{st}^k) d_t^k x_{s\pi t}^k, \tag{33a}$$

$$\text{s.t.} \sum_{\substack{s \in [t] \\ \pi \in [P]^K}} x_{s\pi t}^k = 1, \qquad 1 \le t \le n, k \in [K], \tag{33b}$$

$$0 \le x_{s\pi t}^k \le y_{s\pi}^0, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [P]^K, \tag{33c}$$

$$0 \le x_{s\pi t}^k \le y_{s\pi}^k, \qquad 1 \le s \le t \le n, k \in [K], \pi \in [P]^K, \tag{33d}$$

$$y_{s\pi}^0 \in \{0,1\}, y_{s\pi}^k \in \{0,1\}, \qquad 1 \le s \le n, k \in [K], \pi \in [P]^K. \tag{33e}$$

The intuition behind the generalized JRP is that each time period t in the piecewise-linear concave cost JRP corresponds to P^K time periods (t, π) in the generalized JRP. Each time period (t, π) allows us to use a different combination $\pi = (p_1, \ldots, p_K)$ of pieces of the individual order cost functions $\phi^1, \ldots, \phi^K$.

This formulation does not satisfy the cost assumptions required for the 2-approximation algorithm of Levi et al. [LRS06]. In the next section, we will devise, starting from the algorithm of Levi et al, an algorithm for the generalized JRP that provides a 4-approximation guarantee and runs in exponential time. In Section 4.2, we will employ our technique to obtain a strongly polynomial 4-approximation algorithm for the concave cost JRP.

4.1 An Algorithm for the Generalized JRP

Consider the LP relaxation of IP (33) 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:

$$\max \sum_{k=1}^{K}\sum_{t=1}^{n} v^k_t, \tag{34a}$$

$$\text{s.t. } v^k_t \leq (c^k_{p_k} + h^k_{st})d^k_t + w^k_{s\pi t} + u^k_{s\pi t}, \qquad {\scriptstyle 1\leq s\leq t\leq n, k\in[K],\ \pi\in[P]^K,} \tag{34b}$$

$$\sum_{t=s}^{n} w^k_{s\pi t} \leq f^k_{p_k}, \qquad {\scriptstyle 1\leq s\leq n, k\in[K],\ \pi\in[P]^K,} \tag{34c}$$

$$\sum_{k=1}^{K}\sum_{t=s}^{n} u^k_{s\pi t} \leq f^0, \qquad 1 \leq s \leq n, \pi \in [P]^K, \tag{34d}$$

$$w^k_{s\pi t} \geq 0, u^k_{s\pi t} \geq 0, \qquad {\scriptstyle 1\leq s\leq t\leq n, k\in[K],\ \pi\in[P]^K.} \tag{34e}$$

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

$$w^k_{s\pi t} = \max\{0, v^k_t - (c^k_{p_k} + h^k_{st})d^k_t\} \leq f^k_{p_k}. \tag{35a}$$

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

$$u^k_{s\pi t} = \max\{0, v^k_t - (c^k_{p_k} + h^k_{st})d^k_t - w^k_{s\pi t}\}. \tag{35b}$$

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 individual order (s,k,π) and $w^k_{s\pi t}$ is 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 $u^k_{s\pi t}$ is its contribution.

Since we now have several items, each with its own holding costs, we think of W as a "master" wave, and decrease it from n to 1 and then to a bounded amount below 1. For each item k, we maintain an item wave

$$W^k = h_{1\lfloor W\rfloor} + h_{\lfloor W\rfloor}(W - \lfloor W\rfloor). \tag{36}$$

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

(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 (36). 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 (35). 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 (35). If the joint order (s, π) is also 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, k) 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 individual order (s, k). Serve all demand points (t', k) with $s \le t' \le t$ 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 provides a 4-approximation guarantee. 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].

Theorem 5. *Algorithm* JRPPD *provides a 4-approximation guarantee for the generalized JRP.*

Proof. First, similarly to the proof of Levi et al. and Theorem 3, 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. $\sum_{s=1}^{n} \sum_{\pi \in [P]^K} f^0 y_{s\pi}^0 \le \sum_{k=1}^{K} \sum_{t=1}^{n} v_t^k$. 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 3, 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 individual 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 individual order cost by 3 times the cost of the dual solution, i.e. $\sum_{s=1}^{n}\sum_{\pi\in[P]^K} f_{p_k}^k y_{s\pi}^k + \sum_{s=1}^{n}\sum_{t=s}^{n}\sum_{\pi\in[P]^K}(c_{p_k}^k + h_{st}^k)d_t^k x_{s\pi t}^k \leq 3\sum_{t=1}^{n} v_t^k$.

Therefore, we obtain a 4 approximation algorithm. □

4.2 Applying the Technique

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:

$$\min \sum_{\substack{s\in[n]\\ \pi\in\mathbb{R}_+^K}} f^0 y_{s\pi}^0 + \sum_{\substack{s\in[n],k\in[K]\\ \pi\in\mathbb{R}_+^K}} f_{p_k}^k y_{s\pi}^k + \sum_{\substack{1\leq s\leq t\leq n\\ k\in[K],\pi\in\mathbb{R}_+^K}} (c_{p_k}^k + h_{st}^k)d_t^k x_{s\pi t}^k, \tag{37a}$$

$$\text{s.t.} \sum_{\substack{s\in[t]\\ \pi\in\mathbb{R}_+^K}} x_{s\pi t}^k = 1, \qquad 1\leq t\leq n, k\in[K], \tag{37b}$$

$$0\leq x_{s\pi t}^k \leq y_{s\pi}^0, \qquad 1\leq s\leq t\leq n, k\in[K], \pi\in\mathbb{R}_+^K, \tag{37c}$$

$$0\leq x_{s\pi t}^k \leq y_{s\pi}^k, \qquad 1\leq s\leq t\leq n, k\in[K], \pi\in\mathbb{R}_+^K, \tag{37d}$$

$$y_{s\pi}^0\in\{0,1\}, y_{s\pi}^k\in\{0,1\}, \qquad 1\leq s\leq n, k\in[K], \pi\in\mathbb{R}_+^K. \tag{37e}$$

As before, let ConcaveJRPPD be the algorithm obtained by executing Algorithm JRPPD implicitly on infinitely-sized problem (37).

Theorem 6. *Algorithm* ConcaveJRPPD *is a strongly polynomial 4-approximation algorithm for the concave cost JRP.*

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 $\pi\in\mathbb{R}_+^K$ explicitly. Denote the tangent to the individual ordering cost (ϕ^k, s) that becomes tight first by p_{ks}^*. Then all the other tangents to this individual ordering cost at this time are no longer relevant:

1. Concerning individual 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 $\sum_{k=1}^{K}\sum_{t=s}^{n} u_{s\pi t}^k$ 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 occurs, 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 individual ordering costs (ϕ^k, s). Through this transformation, the wave position when the joint ordering cost becomes tight can be computed by Lemma 6.

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 joint order at time 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. □

Acknowledgments

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

References

[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.

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

[Byr07] Jaroslaw Byrka. An optimal bifactor approximation algorithm for the metric uncapacitated facility location problem. In Moses Charikar, Klaus Jansen, Omer Reingold, and Jos Rolim, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 4627 of *Lecture Notes in Computer Science*, pages 29–43. Springer Berlin / Heidelberg, 2007. 10.1007/978-3-540-74208-1-3.

[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.

[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.

[FLR66] E. Feldman, F. A. Lehrer, and T. L. Ray. Warehouse location under continuous economies of scale. *Management Sci.*, 12(9):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.

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

[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.

[HH61] Warren M. Hirsch and Alan J. Hoffman. Extreme varieties, concave functions, and the fixed charge problem. *Comm. Pure Appl. Math.*, 14:355–369, 1961.

[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.

[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. Hamburger. A heuristic program for locating warehouses. *Management Sci.*, 9(4):643–666, 1963.

[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. Improved approximation algorithm for the one-warehouse 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.

[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.

[MS12] Thomas L. Magnanti and Dan Stratila. Separable concave optimization approximately equals piecewise-linear optimization. Working Paper OR 390-12, Massachusetts Institute of Technology, Operations Research Center, January 2012.

[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.

[PW06] Yves Pochet and Laurence A. Wolsey. *Production planning by mixed integer programming*. Springer Series in Operations Research and Financial Engineering. Springer, New York, 2006.

[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 (El Paso, TX)*, pages 475–484 (electronic). ACM, New York, 1997.

[RSSZ10] H. Edwin Romeijn, Thomas C. Sharkey, Zuo-Jun Max Shen, and Jiawei Zhang. Integrating facility location and production planning decisions. *Networks*, 55(2):78–89, 2010.

[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: citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.4013.

[STA97] David B. Shmoys, Éva Tardos, and Karen Aardal. Approximation algorithms for facility location problems (extended abstract). In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, STOC '97, pages 265–274, New York, NY, USA, 1997. ACM.

[Str08] Dan Stratila. *Combinatorial optimization problems with concave costs*. PhD thesis, Massachusetts Institute of Technology, Operations Research Center, September 2008.

[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.

[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.

[Zan66] W. I. Zangwill. A deterministic multi-product, multi facility production and inventory model. *Operations Research*, 5:89–96, 1966.