A Branch-and-Price Algorithm for Facility Location with General Facility Cost Functions

Wenjun Ni* Jia Shu† Miao Song‡ Dachuan Xu§ and Kaike Zhang¶

March 26, 2021

Abstract

Most existing facility location models assume that the facility cost is either a fixed setup cost or comprised of a fixed setup and a problem-specific concave or submodular cost term. This structural property plays a critical role in developing fast branch-and-price, Lagrangian relaxation, constant ratio approximation, and conic IP reformulation approaches for these NP-hard problems. Many practical considerations and complicating factors, however, can make the facility cost no longer concave or submodular. By removing this restrictive assumption, we study a new location model that considers general nonlinear costs to operate facilities in the facility location framework. The general model does not even admit any approximation algorithms unless $P = NP$ because it takes the unsplittable hard-capacitated metric facility location problem as a special case. We first reformulate this general model as a set-partitioning model and then propose a branch-and-price approach. Although the corresponding pricing problem is NP-hard, we effectively analyze its structural properties and design an algorithm to solve it efficiently. The numerical results obtained from two implementation examples of the general model demonstrate the effectiveness of the solution approach, reveal the managerial implications, and validate the importance to study the general framework.

Key words: Branch-and-Price; Combinatorial Optimization; Facility Location; Integrated Supply Chain

*Department of Management Science and Engineering, School of Economics and Management, Southeast University. Email: wjni@seu.edu.cn

†Department of Management Science and Engineering, School of Economics and Management, Southeast University. Email:jshu@seu.edu.cn

‡Department of Logistics and Maritime Studies, Faculty of Business, The Hong Kong Polytechnic University. Email: miao.song@polyu.edu.hk

§Department of Applied Mathematics, Beijing University of Technology. Email: xudc@bjut.edu.cn.

¶Department of Management Science and Engineering, School of Economics and Management, Southeast University. Email: kaikezhang@utk.edu

1 Introduction

In recent years, a series of important studies demonstrate that facility location models play important roles in strategic supply chain network design. In a comprehensive survey in this area, Melo et al. (2009) point out that the incorporation of facility location models into the supply chain context has opened a very interesting and fruitful research area in both operations research and supply chain management. Over the last couple of decades there is a huge body of literature on location models; see Drezner (1995), Drezner and Hamacher (2002), Eiselt and Marianov (2011), etc. Here we focus on the discrete facility location models (cf. Daskin, 1995). We consider a framework of location-allocation network design and optimization that simultaneously makes the location, transportation, and facility operations decisions. With the aid of this, we can study how the operational decisions can have significant implications on the strategic location decisions in various scenarios.

The basic model in our framework can be described as follows. We are given a set of retail outlets and a set of potential facility locations. Each retailer must be served by exactly one facility to be set up at a potential location. There are three types of costs incurred in this system:

- Fixed cost for opening facilities. This cost is counted on an annual basis.
- Transportation cost. There is a variable cost incurred each time a unit is sent from a facility to a retailer.
- Facility cost. This general nondecreasing cost function is used to capture the facility cost associated with the various attributes of the retailers that the facility serves, including the cost of serving the aggregated demand mean by a facility, the cost incurred by serving the aggregated demand variance of a facility, the routing cost of serving the retailers assigned to a facility, the cost of manufacturing a set of products at a facility, the cost of building sufficient capacity to produce a set of products at a facility, etc.

We would like to determine the optimal locations to set up the facilities, optimally allocate the retailers to the facilities, and optimize the operations strategies for the facilities. The goal is to minimize the total systemwide location, transportation, and facility costs.

If we ignore the facility costs in our basic model, then the problem becomes a classical location problem, which is called the uncapacitated facility location problem (UFLP). The UFLP was originally formulated by Balinski (1965) and Kuehn and Hamburger (1963). Since then a great amount of research has been devoted to this problem; see Cornuéjols et al. (1990) for an excellent review on this problem.

Our modeling framework also expands the applicability of the joint location-inventory model (cf. Daskin et al., 2002 and Shen et al., 2003) by optimizing integrated location-inventory network design for facility operating costs that are general nondecreasing functions of the aggregated demand mean and variance, respectively. In the joint location-inventory model and a series of its generic models (Shen, 2005, Shen and Daskin, 2005, Shen, 2006, Shen and Qi, 2007, Snyder et al., 2007, Sourirajan et al., 2007, Vidyarthi et al., 2007, Naseraldin and Herer, 2008, Mak and Shen, 2009, Park et al., 2010, Qi et al., 2010, Ağralı et al., 2012), these costs are assumed to be concave in the aggreated demand mean or variance served by a facility. Given a finite set I, a concave univariate function $g(\cdot)$, and $b_i \geq 0$ for all $i \in I$, the set function $h(S) = g(\sum_{i \in S} b_i)$, where $S \subseteq I$, is submodular. Thus, the aforementioned costs are submodular functions of the set of retailers the facility serves. In the other important stream of the location-inventory model (cf. Teo and Shu, 2004 and Shu, 2010) that incorporates the infinite horizon two-echelon inventory cost function into the UFLP, the facility operating and two-echelon inventory replenishment costs are also shown to be submodular. The concave and submodular property of these cost functions plays a critical role in designing fast branch-and-price, Lagrangian relaxation, greedy and primal-dual approximation algorithms, and conic integer programming (IP) reformulation approaches for solving this stream of models in the literature.

As shown in Ozsen et al. (2008), however, more realistic consideration, e.g., facility peak level inventory capacity, can lead to a neither concave nor submodular operating cost function. Lu et al. (2014) and the references therein also demonstrate that the more realistic cost components can be non-concave and non-submodular in many important and practical contexts such as facility capacity and congestion cost consideration, less-than-truckload transportation, and nonnested batch ordering. In particular, Lu et al. (2014) propose an interesting location model with an inverse S-shaped cost function to represent the production cost and outline a fast column generation heuristic to address it. Furthermore, in a much earlier work, Desrochers et al. (1995) incorporate a convex congestion cost in a facility location problem. The resulting nonlinear, mixed integer program is solved by column generation and branch-and-bound developed based upon the convexity of the congestion cost. We note that both Lu et al. (2014) and Desrochers et al. (1995) consider a multi-sourcing network, i.e., each retailer can be served by multiple facilities. This is slightly different from the single-sourcing strategy assumed in our model, where each retailer can only be assigned to one facility. As explained in Section 5, our solution approach can be easily adjusted to allow multi-sourcing. The algorithm proposed by Lu et al. (2014), however, may be difficult to be applied to the single-sourcing counterpart. This is because the assignment variables are continuous for multi-sourcing problems. Lu et al. (2014) use this property along with the inverse S-shaped cost to derive the master problem for column generation, which cannot be applied to the single-sourcing counterpart. Also note that neither this master problem nor its dual contains the assignment variables. Consequently, the proposed algorithm cannot be straightforwardly adapted to solve the single-sourcing problem either.

This paper further enhances the practicality of facility location models through removing two restrictions in the literature. First, we relax the concave, submodular, and other problem-specific structural requirements of the facility operating costs. In contrast, we only assume that they are nondecreasing. The general nondecreasing cost terms are extremely useful in modeling production scale-of-(dis)economies, capacity planning costs, inventory costs, and so on. Second, we allow the facility cost function depending on an arbitrary number of attributes, whereas the existing works consider either a concave/submodular cost with at most three attributes (Shen and Qi, 2007) or a non-concave cost with a single attribute (Lu et al., 2014 and Desrochers et al., 1995). Incorporating costs defined by multiple attributes significantly broadens the applicability of our model for studying, e.g., various multi-product facility location problems with nonsubmodular cost functions. We will demonstrate the modeling power of the general model using five concrete applications in Section 2.

Our model is also theoretically very difficult. Assume that the potential facility locations and the retail outlets are in a common metric space. It is easy to see that the model takes the unsplittable hard-capacitated metric facility location model as a special case, for which just deciding whether there exists a feasible solution is NP-complete (cf. Levi et al., 2012) and only bicriteria approximation exists (cf. Bateni and Hajiaghayi, 2012). This directly means that the basic model does not even admit any approximation algorithms unless $P = NP$ (cf. Bateni and Hajiaghayi, 2012). Without an effective approach to solve this model, it is almost impossible to gain any managerial insights on the location and operations decisions in any specific application relevant in practice.

Given these facts, it is thus interesting to develop an effective approach to address this general model. To the best of our knowledge, this is the first time that an algorithm is devised for the general model. Our approach adopts the branch-and-price framework. To be precise, we first recast the general model as a set-partitioning model and propose a branch-and-price algorithm to solve the set-partitioning IP reformulation. We completely characterize the structural properties of the relaxation of the pricing problem, which leads to fast solution of the relaxation and guarantees that the pricing problem can be solved rapidly using the branch-and-bound method. We also provide the computational evidence to demonstrate that the proposed branch-and-price algorithm performs strongly using two applications.

The remainder of this paper is organized as follows. In Section 2, we introduce the location model with general nondecreasing cost functions and its reformulation to facilitate a branch-and-price approach. Section 3 analyzes the structural properties of the pricing problem necessitated by the branch-and-price algorithm to generate new columns and check optimality, and presents the solution algorithm for the pricing problem. In Section 4, we numerically study two applications to demonstrate the effectiveness of the proposed approach. The computational results also shed new insights on the two applications and verify the importance of considering the general facility costs. Finally, the paper is concluded in Section 5. Appendix A of the Online Supplement contains all the proofs.

2 Problem Description and Formulations

In this section, we formally formulate the location model with general nondecreasing facility cost functions and discuss its reformulations. The general model considers a (finite) set J of potential facility locations as well as a (finite) set I of geographically dispersed retailers, each of which faces a customer demand. All the facilities either source the products from an outside supplier or manufacture them by themselves and the demand of each retailer is fulfilled from a single facility. The problem is to determine (i) which set of facilities to open, (ii) the assignment of facilities to retailers (assuming single-sourcing), and (iii) the operations strategy at each facility, so as to minimize the total systemwide location, transportation, and other operating costs. To facilitate the model development, we first define the following notation.

Sets

- J: set of potential facility locations
- I: set of retailers, $|I| = n$

Cost Parameters and Functions

- F_j: fixed (yearly) cost of operating a facility at location j, $\forall j \in J$

- t_{ij}: linear transportation cost to ship to retailer i from facility j, $\forall i \in I, j \in J$

- b_{ij}^k: k-th nonnegative attribute at retailer i if it is served by facility j, $\forall i \in I, j \in J, k \in \{1, 2, \ldots, \kappa\}$, which is introduced to model the facility cost. For example, an attribute could represent the demand mean, the demand variance, the constant demand rate, or the weighted shipping distance of product k from facility j to retailer i via routing

- $\Gamma_j\Big(\sum_{i\in S} b_{ij}^1, \sum_{i\in S} b_{ij}^2, \cdots, \sum_{i\in S} b_{ij}^\kappa\Big)$: facility cost incurred by facility j for serving retailers in $S \subseteq I$, which is nondecreasing, $\forall j \in J$. It comprises the costs from serving the κ attributes of the retailers in S

Decision Variables

$$X_j = \begin{cases} 1, & \text{if facility } j \text{ is open;} \\ 0, & \text{otherwise,} \end{cases} \quad \text{and} \quad Y_{ij} = \begin{cases} 1, & \text{if retailer } i \text{ is served by facility } j; \\ 0, & \text{otherwise.} \end{cases}$$

We are now ready to formulate the location model with general nondecreasing facility cost functions as follows:

$$\begin{aligned} \mathcal{P}: \quad \min \quad & \sum_{j\in J}\left(F_j X_j + \sum_{i\in I} t_{ij} Y_{ij} + \Gamma_j\left(\sum_{i\in I} b_{ij}^1 Y_{ij}, \sum_{i\in I} b_{ij}^2 Y_{ij}, \cdots, \sum_{i\in I} b_{ij}^\kappa Y_{ij}\right)\right) \\ \text{s.t.} \quad & \sum_{j\in J} Y_{ij} = 1, \qquad \forall i \in I, \\ & Y_{ij} - X_j \le 0, \quad \forall i \in I,\ j \in J, \\ & Y_{ij} \in \{0,1\}, \qquad \forall i \in I,\ j \in J, \\ & X_j \in \{0,1\}, \qquad \forall j \in J, \end{aligned}$$

where the objective function minimizes the costs of locating facilities, transporting to retailers via facilities, and serving the retailers. The constraints in $\mathcal{P}$ are the standard constraints same as those in the UFLP. We note that when $\kappa = 1$ and $\Gamma_j(\cdot)$ is a concave function of the aggregated demand mean served by facility j, $\mathcal{P}$ reduces to the joint location-inventory model that can be rapidly solved, respectively, by column generation (Shen et al., 2003) and Lagrangian relaxation (Daskin et al., 2002). When $\kappa = 2$ and $\Gamma_j(\cdot,\cdot)$ is a sum of two concave functions of the aggregated demand mean and variance served by facility j, respectively, $\mathcal{P}$ becomes the stochastic transportation-inventory network design problem (STIND) studied in Shu et al. (2005), in which a fast column generation algorithm is developed to address it. In both special cases, $\mathcal{P}$ can be viewed as facility location problems with monotone submodular facility costs. If the set of potential facility locations and the set of retail outlets are in a common metric space, there also exist strongly polynomial-time greedy 1.861-approximation and primal-dual 3-approximation algorithms for $\mathcal{P}$ under these two scenarios, respectively (cf. Hajiaghayi et al., 2003 and Li et al., 2013). When $\kappa = 2$ and $\Gamma_j(\cdot, \cdot)$ is a sum of square root functions each corresponding to an attribute, Atamtürk et al. (2012) propose a conic IP approach that can solve $\mathcal{P}$ rapidly. These fast solution approaches all rely on the concave or submodular property of the facility cost functions. As shown in Section 1, many practical considerations, however, may result in a non-concave or non-submodular Γ_j. Further complicated by the arbitrariness of κ, $\mathcal{P}$ is significantly more difficult to solve than the previous models.

Model $\mathcal{P}$ takes many well-established facility location models as special cases. Obviously, it is reduced to the UFLP if we remove the terms Γ_j in $\mathcal{P}$. Some practical applications within the modeling framework are listed as follows.

Application 1. Capacitated/Congested Facility Location Problem. Let $\kappa = 1$ and b_i^1 represent the constant demand rate of retailer i. The capacitated facility location model corresponds to the case with

$$\Gamma_j\left(\sum_{i\in S} b_i^1\right) = \begin{cases} 0, & \sum_{i\in S} b_i^1 \le C_j, \\ +\infty, & \sum_{i\in S} b_i^1 > C_j, \end{cases}$$

where C_j denotes the capacity at facility j. The congested single-sourcing facility problem also falls into our framework if $\Gamma_j(\sum_{i\in S} b_i^1)$ is a convex function representing the congestion cost for facility j to serve the aggregated demand $\sum_{i\in S} b_i^1$. For example, in Desrochers et al. (1995), the congestion cost for facility j to serve a total demand of q_j is set to $\Gamma_j(q_j) = f_j(q_j) \times q_j$, where $f_j(\cdot)$ is a nonnegative, increasing, continuous, and convex function.

Application 2. Joint Location-Inventory Model. Consider the case with $\kappa = 2$. Let b_i^1 and b_i^2 denote the mean and variance of the demand of a single product at retailer i, respectively. Then the facility cost $\Gamma_j(\cdot, \cdot)$ can be interpreted as the sum of two cost terms, i.e., $\Gamma_j^1(\sum_{i\in S} b_i^1)$ and $\Gamma_j^2(\sum_{i\in S} b_i^2)$, determined by the mean and variance of the total demand served by facility j. The joint location-inventory model (e.g., Daskin et al., 2002, Shen et al., 2003, and Shu et al., 2005) can be cast into this framework, in which both $\Gamma_j^1(\cdot)$ and $\Gamma_j^2(\cdot)$ are concave. The location-inventory network design with routing costs proposed by Shen and Qi (2007) is also a special case of model $\mathcal{P}$ with $\kappa = 3$, where $\Gamma_j(\cdot, \cdot, \cdot)$ is a sum of the aforementioned $\Gamma_j^1(\sum_{i\in S} b_i^1)$ and $\Gamma_j^2(\sum_{i\in S} b_i^2)$ as well as another concave cost term $\Gamma_j^3(\sum_{i\in S} b_{ij}^3)$. Here b_{ij}^3 is the third attribute of retailer i defined in a way such that $\sum_{i\in S} b_{ij}^3$ approximates the routing distance from facility j to the set of retailers S served by facility j. While the previous works all assume the concavity of $\Gamma_j^1(\cdot)$, $\Gamma_j^2(\cdot)$, and $\Gamma_j^3(\cdot)$, model $\mathcal{P}$ allows them being arbitrarily nondecreasing functions, which can easily model additional practical issues such as peak inventory level capacity (cf. Ozsen et al., 2008) as well as non-concave routing and congestion costs.

Application 3. Location and Production Model. Consider a multi-product facility location model where each retailer i faces constant demands of κ types of products and the facilities manufacture the κ products to satisfy the demand. Let b_i^k be the demand rate of product k at retailer i. The cost for facility j to produce q_j^k units of product k is a general nonlinear function $\Gamma_j^k(q_j^k)$. Then model $\mathcal{P}$ with $\Gamma_j(\sum_{i\in S} b_i^1, \sum_{i\in S} b_i^2, \cdots, \sum_{i\in S} b_i^\kappa) = \sum_{k=1}^{\kappa} \Gamma_j^k(\sum_{i\in S} b_i^k)$ formulates the facility location problem to minimize the total cost of location, transportation, and production. The model studied in Lu et al. (2014) can be viewed as the multi-sourcing counterpart with $\kappa = 1$.

Application 4. Location and Capacity Planning Model. Similar to Application 3, we consider a set of retailers facing demands of m products manufactured by the facilities. Let D_i^k denote the stochastic demand rate of product k at retailer i. Suppose that the fill rate of each product at each retailer should be at least $1-\epsilon$, where $\epsilon \in (0,1)$ is a small positive number. In other words, if facility j serves the set of retailers in S, then its production capacity q_j^k of product k should satisfy $P(q_j^k \geq \sum_{i\in S} D_i^k) \geq 1-\epsilon$. We can determine the value of q_j^k under various conditions:

- If the demand rate D_i^k for any $i \in S$ follows an independent normal distribution with mean a_i^k and variance b_i^k, then

  $$q_j^k = \sum_{i\in S} a_i^k + F^{-1}(1-\epsilon)\sqrt{\sum_{i\in S} b_i^k},$$

  where $F^{-1}(\cdot)$ denotes the inverse of the standard normal cumulative distribution function.

- Suppose that D_i^k for any $i \in S$ is independently distributed with mean a_i^k and variance b_i^k. According to the one-sided Chebyshev's inequality, we obtain

  $$q_j^k = \sum_{i\in S} a_i^k + \sqrt{\frac{1-\epsilon}{\epsilon}\sum_{i\in S} b_i^k}.$$

- Suppose that the demand rate D_i^k for any $i \in S$ is independently distributed with mean a_i^k

and support $\left[\underline{b}_i^k, \underline{b}_i^k + \sqrt{b_i^k}\right]$. Hoeffding's inequality yields that

$$q_j^k = \sum_{i \in S} a_i^k + \sqrt{\frac{-\ln \epsilon}{2} \sum_{i \in S} b_i^k}.$$

In the above three cases, we can represent q_j^k as $\sum_{i \in S} a_i^k + Q_j^k(\sum_{i \in S} b_i^k)$. Suppose that the cost to build the capacity q_j^k for product k at facility j is $\Gamma_j^k(q_j^k)$. Then

$$\Gamma_j\Big(\sum_{i \in S} a_i^1, \sum_{i \in S} b_i^1, \sum_{i \in S} a_i^2, \sum_{i \in S} b_i^2, \cdots, \sum_{i \in S} a_i^m, \sum_{i \in S} b_i^m\Big) = \sum_{k=1}^{m} \Gamma_j^k \left(\sum_{i \in S} a_i^k + Q_j^k\Big(\sum_{i \in S} b_i^k\Big)\right)$$

in model $\mathcal{P}$ corresponds to the total cost for facility j to get sufficient capacity to serve the retailers in the set S. In this case, we have $\kappa = 2m$ as the attributes are a_i^k and b_i^k for all $k \in \{1, ..., m\}$.

In order to obtain a linearized formulation, $\mathcal{P}$ is reformulated as the following set-partitioning model:

$$\begin{aligned} \min \quad & \sum_{j \in J} \sum_{S \subseteq I} C_{j,S} X_{j,S} \\ \text{s.t.} \quad & \sum_{j \in J} \sum_{S \subseteq I : i \in S} X_{j,S} = 1, \quad \forall i \in I, \\ & \sum_{S \subseteq I} X_{j,S} = 1, \qquad\qquad \forall j \in J, \\ & X_{j,S} \in \{0, 1\}, \qquad\qquad \forall j \in J,\ S \subseteq I. \end{aligned} \tag{1}$$

Here, $C_{j,S}$ represents the total location, transportation, and facility cost for facility j to serve the retailers in S, i.e.,

$$C_{j,S} = \mathbb{1}_{S \neq \emptyset} F_j + \sum_{i \in S} t_{ij} + \Gamma_j\Big(\sum_{i \in S} b_{ij}^1, \sum_{i \in S} b_{ij}^2, \cdots, \sum_{i \in S} b_{ij}^{\kappa}\Big) \text{ where } \mathbb{1}_{S \neq \emptyset} = \begin{cases} 1, & \text{if } S \subseteq I \text{ and } S \neq \emptyset, \\ 0, & \text{if } S = \emptyset. \end{cases}$$

Moreover, $X_{j,S}$ is the decision variable that equals 1 if facility j serves the retailers in S and 0 otherwise. Consequently, the original nonlinear objective in $\mathcal{P}$ is linearized at the expense of introducing an exponential number of variables ($O(2^n)$), which justifies the use of branch-and-price to solve (1).

To develop the branch-and-price algorithm, we first construct the pricing problem for the column generation procedure that solves the linear program (LP) at the root node of the branch-and-price tree, i.e., the LP relaxation of (1) obtained by relaxing the integrality constraints. Obviously, each pair of (j, S) corresponds to a decision variable $X_{j,S}$ and forms a column. Consider the following dual of the LP relaxation of (1):

$$\begin{aligned}\max \quad & \sum_{i\in I}\xi_i + \sum_{j\in J}\eta_j \\ \text{s.t.} \quad & \sum_{i\in S}\xi_i + \eta_j \le C_{j,S}, \quad \forall j\in J,\ S\subseteq I,\end{aligned}$$

in which ξ_i for all $i \in I$ and η_j for all $j \in J$ denote the dual variables corresponding to the first and second constraints of (1), respectively. Let $(\bar{\xi}_i, \bar{\eta}_j\, \forall i \in I, j \in J)$ be the dual solution obtained in an iteration of the column generation algorithm to solve the LP relaxation of (1). For any column (j, S), the reduced cost is calculated by

$$\bar{C}_{j,S} = \mathbb{1}_{S\neq\emptyset}F_j + \sum_{i\in S}(t_{ij} - \bar{\xi}_i) + \Gamma_j\left(\sum_{i\in S} b^1_{ij}, \sum_{i\in S} b^2_{ij}, \cdots, \sum_{i\in S} b^\kappa_{ij}\right) - \bar{\eta}_j.$$

Consider any given $j \in J$. We would like to know if the reduced costs for all columns (j, S) are nonnegative. Define

$$\bar{C}^{\min}_j = \min\left\{F_j + \bar{C}_{j,\emptyset},\ \min_{S\subseteq I: S\neq\emptyset} \bar{C}_{j,S}\right\}. \tag{2}$$

Suppose that the decision variable $X_{j,\emptyset}$ has been considered in the initial columns of the column generation procedure, which ensures $F_j + \bar{C}_{j,\emptyset} \ge \bar{C}_{j,\emptyset} \ge 0$. Then $\bar{C}^{\min}_j < 0$ if and only if $\min_{S\subseteq I:S\neq\emptyset} \bar{C}_{j,S} < 0$. In other words, if $\bar{C}^{\min}_j \ge 0$, all columns (j, S) have nonnegative reduced costs; if $\bar{C}^{\min}_j < 0$, there exists a column (j, S^*) with a negative reduced cost, where $S^* \neq \emptyset$ is an optimal solution to model (2).

Apparently, model (2) is equivalent to the following binary optimization problem $\mathcal{SP}_j$:

$$\begin{aligned}\mathcal{SP}_j: \quad \min \quad & F_j + \sum_{i\in I}(t_{ij} - \bar{\xi}_i)Y_{ij} + \Gamma_j\left(\sum_{i\in I} b^1_{ij}Y_{ij}, \sum_{i\in I} b^2_{ij}Y_{ij}, \cdots, \sum_{i\in I} b^\kappa_{ij}Y_{ij}\right) - \bar{\eta}_j \\ \text{s.t.} \quad & Y_{ij} \in \{0,1\}, \quad \forall i \in I,\end{aligned}$$

which is called the pricing problem for the branch-and-price algorithm at the root node. At any other node of the branch-and-price tree, the pricing problem is still in the form of $\mathcal{SP}_j$ as long as we branch on $\sum_{S\subseteq I:i\in S} X_{j,S}$ for some $i \in I$ and $j \in J$, i.e., whether retailer i is served by j. Please refer to the discussion in Section 4.1.2 of Barnhart et al. (1998) and the references therein. We also note that Lagrangian relaxation can be employed to solve the formulation $\mathcal{P}$ and the same subproblem $\mathcal{SP}_j$ must be solved in each iteration. Therefore, the fast implementation of both branch-and-price and Lagrangian relaxation depends on whether we can solve $\mathcal{SP}_j$ rapidly.

When $\kappa = 1$ and $\Gamma_j(\cdot)$ is concave, Shen et al. (2003) propose an $O(n \log n)$ algorithm for the pricing problem. When $\kappa = 2$ and $\Gamma_j(\cdot, \cdot)$ is a sum of two single-variate concave functions, Shu et al.

(2005) show that the corresponding pricing problem can be solved in $O(n^2 \log n)$. The concavity of Γ_j ensures a parametric linear programming representation of the problem for which optimality can be attained at an extreme point. This results in an efficient special-purpose strongly polynomial-time algorithm for such a pricing problem. As a matter of fact, under these special cases, the pricing problem is a submodular function minimization problem and hence is readily solvable in polynomial-time. Unfortunately, when the concavity assumption is relaxed, such a property no longer holds. As shown in the next section, the general problem $\mathcal{SP}_j$ is NP-hard. However, we can still exploit structural properties that lead to its fast solution.

Application 5. Pricing Problem Model. Although the pricing problem is derived from the location model with general facility cost functions, it is likely to have important applications in other areas as well. Many problems, such as the performance achievability checking problem for multiclass queueing systems (cf. Federgruen and Groenevelt, 1988), the market selection problem (cf. Geunes et al., 2004), the operating room scheduling problem (cf. Lamiri et al., 2008), and the selective newsvendor problem (cf. Strinka et al., 2013), are special cases of the pricing problem $\mathcal{SP}_j$. Although all the aforementioned models possess a submodular structure, a simple practical consideration, e.g., capacity, may destroy the property. Therefore, there are potentially many important applications of this work that go well beyond location-allocation network design modeling.

3 The Pricing Problem

With straightforward changes of notation, we can transform the pricing problem $\mathcal{SP}_j$ for any $j \in J$ into the following form:

$$\mathcal{Q}: \quad \min \left\{ -\sum_{i=1}^{n} a_i z_i + \Phi\left(\sum_{i=1}^{n} \mathbf{b}_i z_i\right) \;\middle|\; z_i \in \{0,1\} \; \forall i \in I \right\}$$

where the function $\Phi : \mathbb{R}_+^\kappa \mapsto \mathbb{R}$ is assumed to be nondecreasing, i.e., $\Phi(\mathbf{x}) \leq \Phi(\mathbf{x}')$ for any $\mathbf{0} \leq \mathbf{x} \leq \mathbf{x}'$. Here, a_i and $\mathbf{b}_i = (b_i^1, b_i^2, ..., b_i^\kappa)^T$ are given constants and $\mathbf{b}_i \geq \mathbf{0}$. We can assume $a_i > 0$ for any $i \in I$ without loss of generality, since otherwise such i will correspond to $z_i^* = 0$ in an optimal solution. Similarly, we can also assume that $\max\{b_i^k : k = 1, ..., \kappa\} > 0$, i.e., $\mathbf{b}_i \neq \mathbf{0}$, for each $i \in I$ since otherwise an optimal solution should have $z_i^* = 1$.

$\mathcal{Q}$ is a very challenging problem as it has a non-convex non-concave objective function and $|I|$ binary decision variables. As a matter of fact, it is NP-hard even when $\kappa = 1$.

Theorem 1. *$\mathcal{Q}$ is NP-hard even if $\kappa = 1$.*

Remark: Theorem 1 directly yields that the pricing problem for the continuous-time single-sourcing problem proposed in Huang et al. (2005) and the Lagrangian relaxation subproblem for the capacitated warehouse location problem with risk pooling proposed in Ozsen et al. (2008) are NP-hard. Both Huang et al. (2005) and Ozsen et al. (2008) prove that a simple greedy allocation rule can solve the continuous relaxation of $\mathcal{Q}$ with $\kappa = 1$.

Despite the NP-hardness, any optimal solution of $\mathcal{Q}$ has the following property, which could potentially speed up a branch-and-bound scheme for the solution of $\mathcal{Q}$ by fixing certain values of z_i after branching.

Proposition 1. *For any optimal solution $\mathbf{z}^*$ to $\mathcal{Q}$, (i) if $z^*_{i^*} = 0$ for some $i^* \in I$, then $z^*_i = 0$ for any $i \in I$ such that $a_i < a_{i^*}$ and $\mathbf{b}_i \geq \mathbf{b}_{i^*}$; (ii) if $z^*_{i^*} = 1$ for some $i^* \in I$, then $z^*_i = 1$ for any $i \in I$ such that $a_i > a_{i^*}$ and $\mathbf{b}_i \leq \mathbf{b}_{i^*}$.*

Furthermore, the *relaxation* of $\mathcal{Q}$, where the binary constraint is relaxed to $z_i \in [0, 1]$ for all $i \in I$, also has very interesting structural properties, which, along with Proposition 1, permit a rapidly implementable branch-and-bound (B&B) algorithm for solving $\mathcal{Q}$. The remainder of the section is devoted to the analysis of the relaxation of $\mathcal{Q}$.

3.1 Number of Fractional Components

First of all, we study the number of fractional components in an optimal solution to the relaxation of $\mathcal{Q}$. Let $\hat{\kappa}$ denote the maximum number of linearly independent vectors in the set $\{\mathbf{b}_i : i \in I\}$, i.e.,

$$\hat{\kappa} = \text{rank}\Big([\mathbf{b}_1\, \mathbf{b}_2\, \cdots\, \mathbf{b}_n]\Big) \leq \kappa. \tag{3}$$

Without loss of generality, we assume that the first $\hat{\kappa}$ rows of the matrix $[\mathbf{b}_1\, \mathbf{b}_2\, \cdots\, \mathbf{b}_n]$ are linearly independent. Then the remaining rows can be represented as a linear combination of the first $\hat{\kappa}$ rows. In other words, there exist $\hat{\kappa}$ linearly independent vectors in the set

$$\left\{\hat{\mathbf{b}}_i \equiv [b^1_i, b^2_i, ..., b^{\hat{\kappa}}_i]^T : i \in I\right\}. \tag{4}$$

For any $k \in \{\hat{\kappa}+1, ..., \kappa\}$, there exists a vector $\mathbf{p}_k \in \mathbb{R}^{\hat{\kappa}}$ such that

$$b^k_i = \mathbf{p}^T_k \hat{\mathbf{b}}_i \quad \forall i \in I. \tag{5}$$

Obviously, $\mathbf{b}_i \neq \mathbf{0}$ implies $\hat{\mathbf{b}}_i \neq \mathbf{0}$ for any $i \in I$.

The following theorem shows that the number of fractional components is at most $\hat{\kappa}$.

Theorem 2. *There exists an optimal solution to the relaxation of $\mathcal{Q}$ that contains at most $\hat{\kappa}$ fractional components.*

If $\Phi(\cdot)$ is quasi-concave, we can further prove that the relaxation of $\mathcal{Q}$ has an optimal solution with at most one fractional component.

Proposition 2. *If $\Phi(\cdot)$ is a quasi-concave function, then there exists an optimal solution to the relaxation of $\mathcal{Q}$ that contains at most one fractional component.*

Theorem 2 and Proposition 2 show that the relaxation of $\mathcal{Q}$ has an optimal solution with a small number of fractional components. Therefore, as long as the relaxation of $\mathcal{Q}$ can be solved rapidly, a simple B&B algorithm, e.g., Algorithm 3 in Appendix B of the Online Supplement, can be developed based on Proposition 1 to solve $\mathcal{Q}$ rapidly. We study how to solve the relaxation of $\mathcal{Q}$ in Sections 3.2 and 3.3.

3.2 Linear Partitioning of the Retailers

The following proposition presents a neat structural property for any optimal solution to the relaxation of $\mathcal{Q}$, which is crucial to rapidly solve the problem.

Proposition 3. *For any optimal solution ($z_i^* \, \forall i \in I$) to the relaxation of $\mathcal{Q}$, there exists $\boldsymbol{\alpha} \geq \mathbf{0}$ such that (i) $\boldsymbol{\alpha}^T \mathbf{b}_i / a_i \leq 1$ for any i such that $z_i^* = 1$, (ii) $\boldsymbol{\alpha}^T \mathbf{b}_i / a_i = 1$ for any i such that $z_i^* \in (0,1)$, and (iii) $\boldsymbol{\alpha}^T \mathbf{b}_i / a_i \geq 1$ for any i such that $z_i^* = 0$.*

To interpret Proposition 3, we can map retailer i with the parameters $(a_i, \mathbf{b}_i)$ to the point $\mathbf{b}_i / a_i$ in a κ-dimensional space. Any optimal solution to the relaxation of $\mathcal{Q}$ corresponds to a partition of the set I by a hyperplane $\{\mathbf{x} \in \mathbb{R}^\kappa : \boldsymbol{\alpha}^T \mathbf{x} = 1\}$, where $\boldsymbol{\alpha} \geq 0$. Given such a vector $\boldsymbol{\alpha}$, we can obtain the following three sets:

$$S_1(\boldsymbol{\alpha}) = \{i \in I : \boldsymbol{\alpha}^T \mathbf{b}_i / a_i < 1\}, \;\; S_0(\boldsymbol{\alpha}) = \{i \in I : \boldsymbol{\alpha}^T \mathbf{b}_i / a_i > 1\}, \;\text{ and }\; H(\boldsymbol{\alpha}) = I \setminus S_1(\boldsymbol{\alpha}) \setminus S_0(\boldsymbol{\alpha}). \tag{6}$$

The retailers on one side of the hyperplane (not inclusive), i.e., in the set $S_1(\boldsymbol{\alpha})$, should be served, while those on the other side of the hyperplane (not inclusive), i.e., in the set $S_0(\boldsymbol{\alpha})$, should not be served. We only need to optimize the values of z_i for the retailers on the hyperplane, i.e., in the set $H(\boldsymbol{\alpha})$, which could be fractionally served. This observation leads to Algorithm 1 that decomposes the relaxation of $\mathcal{Q}$ to problems with reduced numbers of decision variables.

Algorithm 1: Solving the relaxation of $\mathcal{Q}$

Data: $a_i \in \mathbb{R}_+$ and $\mathbf{b}_i \in \mathbb{R}_+^\kappa$ for all $i \in I$, $\hat{\kappa}$ in (3), and a nondecreasing function $\Phi : \mathbb{R}_+^\kappa \mapsto \mathbb{R}$

Result: an optimal solution ($z_i^* \; \forall i \in I$) and the optimal value v^* of the relaxation of $\mathcal{Q}$

1 let $z_i^* := 0$ for all $i \in I$, $v^* := \Phi(\mathbf{0})$, and $A := \emptyset$;

2 **foreach** $T \subseteq I$ *such that* $|T| = \hat{\kappa}$ *and* $\{\hat{\mathbf{b}}_i : i \in T\}$ *are linearly independent* **do**

3 let $\boldsymbol{\beta} := \left[\hat{\boldsymbol{\beta}}^T \, \mathbf{0}^T\right]^T \in \mathbb{R}^\kappa$ where $\hat{\boldsymbol{\beta}}$ satisfies $\hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i = 1$ for all $i \in T$;

4 **if** $\hat{\boldsymbol{\beta}} \in A$ **then** go to Line 13;

5 compute $S_0(\boldsymbol{\beta})$, $S_1(\boldsymbol{\beta})$, and $H(\boldsymbol{\beta})$ defined in (6);

6 **if** $|H(\boldsymbol{\beta})| > \hat{\kappa}$ **then** $A := A \cup \{\hat{\boldsymbol{\beta}}\}$;

7 **if** $\boldsymbol{\beta} \ngeq \mathbf{0}$ **then**

8 solve the following linear program

$$\mathcal{P}(\boldsymbol{\beta}) : \quad \max_{\substack{\boldsymbol{\alpha}, \epsilon \\ \epsilon_i \forall i \in S_1(\boldsymbol{\beta}) \\ \epsilon_i \forall i \in S_0(\boldsymbol{\beta})}} \left\{ \epsilon \;\middle|\; \begin{array}{l} \boldsymbol{\alpha}^T \mathbf{b}_i / a_i + \epsilon_i = 1 \; \forall i \in S_1(\boldsymbol{\beta}), \; \boldsymbol{\alpha} \geq \mathbf{0}, \\ \boldsymbol{\alpha}^T \mathbf{b}_i / a_i - \epsilon_i = 1 \; \forall i \in S_0(\boldsymbol{\beta}), \; \epsilon \leq \epsilon_i \; \forall i \in S_1(\boldsymbol{\beta}) \cup S_0(\boldsymbol{\beta}) \end{array} \right\},$$

and let ϵ^* be the corresponding optimal value;

9 **if** $\epsilon^* \leq 0$ **then** go to Line 13;

10 **end**

11 solve the following optimization problem

$$\mathcal{R}(\boldsymbol{\beta}) : \quad \min \left\{ -\sum_{i \in H(\boldsymbol{\beta})} a_i z_i + \Phi\left(\sum_{i \in H(\boldsymbol{\beta})} \mathbf{b}_i z_i + \sum_{i \in S_1(\boldsymbol{\beta})} \mathbf{b}_i \right) \;\middle|\; z_i \in [0,1] \; \forall i \in H(\boldsymbol{\beta}) \right\},$$

and let ($\bar{z}_i \; \forall i \in H(\boldsymbol{\beta})$) and $\bar{v}$ denote its optimal solution and optimal value, respectively;

12 **if** $\bar{v} - \sum_{i \in S_1(\boldsymbol{\beta})} a_i < v^*$ **then** let $v^* := \bar{v} - \sum_{i \in S_1(\boldsymbol{\beta})} a_i$, $z_i^* := \bar{z}_i$ for all $i \in H(\boldsymbol{\beta})$, $z_i^* := 1$ for all $i \in S_1(\boldsymbol{\beta})$, and $z_i^* := 0$ for all $i \in S_0(\boldsymbol{\beta})$;

13 **end**

14 **return** ($z_i^* \; \forall i \in I$) and v^*;

Proposition 4. *Algorithm 1 solves the relaxation of $\mathcal{Q}$.*

The basic idea of Algorithm 1 is to find a partition of I corresponding to an optimal solution by enumeration. Although any $\boldsymbol{\alpha} \geq \mathbf{0}$, of which there are infinitely many, generates a partition, we can show that it is sufficient to consider the partitioning hyperplanes passing through $\hat{\kappa}$ linearly independent vectors in the set $\{\mathbf{b}_i/a_i : i \in I\}$, where $\hat{\kappa}$ in (3) denotes the number of linearly independent vectors in $\{\mathbf{b}_i : i \in I\}$. These partitions correspond to the vectors $\boldsymbol{\beta}$ generated in Line 3 of Algorithm 1. Thus, Algorithm 1 enumerates at most $C_n^{\hat{\kappa}}$ number of $\boldsymbol{\beta}$ and solves $\mathcal{P}(\boldsymbol{\beta})$ (and $\mathcal{R}(\boldsymbol{\beta})$) for each $\boldsymbol{\beta}$. $\mathcal{P}(\boldsymbol{\beta})$ is a linear program and hence can be solved efficiently. However, $\mathcal{R}(\boldsymbol{\beta})$ may remain challenging especially for large $|H(\boldsymbol{\beta})|$. Therefore, in Section 3.3, we further decompose $\mathcal{R}(\boldsymbol{\beta})$ to problems with $\hat{\kappa}$ decision variables.

3.3 Retailers on the Partitioning Hyperplane

Consider the problem $\mathcal{R}(\boldsymbol{\beta})$ defined in Line 11 of Algorithm 1. Note that $\boldsymbol{\beta}^T\mathbf{b}_i/a_i = 1$, i.e., $a_i = \boldsymbol{\beta}^T\mathbf{b}_i$, for all $i \in H(\boldsymbol{\beta})$. Thus, $\mathcal{R}(\boldsymbol{\beta})$ can be written as

$$\min\left\{ -\sum_{i\in H(\boldsymbol{\beta})} \boldsymbol{\beta}^T\mathbf{b}_i z_i + \Phi\left(\sum_{i\in H(\boldsymbol{\beta})}\mathbf{b}_i z_i + \sum_{i\in S_1(\boldsymbol{\beta})}\mathbf{b}_i\right) \,\middle|\, z_i \in [0,1]\ \forall i \in H(\boldsymbol{\beta})\right\}.$$

Define $\hat{\Phi} : \mathbb{R}^{\kappa} \times \mathbb{R}_+^{\hat{\kappa}} \mapsto \mathbb{R}$ such that

$$\hat{\Phi}(\boldsymbol{\beta},\mathbf{x}) \equiv \Phi\left(\left[x^1,...,x^{\hat{\kappa}},\mathbf{p}_{\hat{\kappa}+1}^T\mathbf{x},...,\mathbf{p}_{\kappa}^T\mathbf{x}\right]^T + \sum_{i\in S_1(\boldsymbol{\beta})}\mathbf{b}_i\right) - \boldsymbol{\beta}^T\left[x^1,...,x^{\hat{\kappa}},\mathbf{p}_{\hat{\kappa}+1}^T\mathbf{x},...,\mathbf{p}_{\kappa}^T\mathbf{x}\right]^T \quad (7)$$

for any $\boldsymbol{\beta} \in \mathbb{R}^{\kappa}$ and $\mathbf{x} = (x^1, x^2, ..., x^{\hat{\kappa}})^T \in \mathbb{R}_+^{\hat{\kappa}}$, where $\mathbf{p}_k$ is introduced in (5). For any set $S \subseteq I$ and function $\Psi : \mathbb{R}_+^{\hat{\kappa}} \mapsto \mathbb{R}$, we can consider the optimization problem $\mathcal{P}(S, \Psi(\cdot))$ defined as

$$\mathcal{P}(S,\Psi(\cdot)): \quad \min\left\{\Psi\left(\sum_{i\in S}\hat{\mathbf{b}}_i z_i\right) \,\middle|\, z_i \in [0,1]\ \forall i \in S\right\}. \quad (8)$$

Given $H(\boldsymbol{\beta})$ in (6) and $\hat{\Phi}(\boldsymbol{\beta},\cdot)$ in (7), $\mathcal{R}(\boldsymbol{\beta})$ is then equivalent to the problem $\mathcal{P}(H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta},\cdot))$.

Obviously, the objective value of $\mathcal{P}(S, \Psi(\cdot))$ is determined by $\sum_{i\in S}\hat{\mathbf{b}}_i z_i$. The following proposition presents a property regarding $\sum_{i\in S}\hat{\mathbf{b}}_i z_i$ where $z_i \in [0, 1]$ for all $i \in S$, which can be applied to decompose $\mathcal{P}(S, \Psi(\cdot))$ into problems with $\hat{\kappa}$ decision variables.

Proposition 5. *Consider any $i^* \in S \subseteq I$ and $\bar{z}_i \in [0, 1]$ for all $i \in S$. There exist $z_i \in [0, 1]$ for all $i \in S$ with $\sum_{i\in S}\hat{\mathbf{b}}_i z_i = \sum_{i\in S}\hat{\mathbf{b}}_i\bar{z}_i$, which satisfy at least one of the following two conditions: (i)*

$z_{i^*} = 1$; or (ii) there exists $\boldsymbol{\theta} \in \mathbb{R}^{\hat{\kappa}}$ such that $\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} > 0$, $z_i = 0$ for all $i \in S \setminus \{i^*\}$ and $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i > 0$, and $z_i = 1$ for all $i \in S \setminus \{i^*\}$ and $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0$.

If $\hat{\mathbf{b}}_{i^*}$ cannot be represented as a linear combination of $\{\hat{\mathbf{b}}_i : i \in S \setminus \{i^*\}\}$, then there exist $z_i \in [0, 1]$ for all $i \in S$ with $\sum_{i \in S} \hat{\mathbf{b}}_i z_i = \sum_{i \in S} \hat{\mathbf{b}}_i \bar{z}_i$ and satisfying the condition (ii).

Remark: $\sum_{i \in S} \hat{\mathbf{b}}_i z_i$ where $z_i \in [0, 1]$ for all $i \in S$ is a special linear combination of $\{\hat{\mathbf{b}}_i : i \in S\}$ with interesting applications, e.g., the linear relaxation of the multi-dimensional knapsack problem. Therefore, Proposition 5 may have other potential applications.

Similar to the interpretation of Proposition 3, we can map any retailer $i \in S$ to the point $\hat{\mathbf{b}}_i$ in a $\hat{\kappa}$-dimensional space. Now consider any feasible solution to $\mathcal{P}(S, \Psi(\cdot))$ and an arbitrary $i^* \in S$. According to Proposition 5, it is possible to get an equivalent feasible solution, i.e., one with the same objective function value, by setting $z_{i^*} = 1$. If it is impossible to do so, then we can find a partitioning hyperplane $\{\mathbf{x} \in \mathbb{R}^{\hat{\kappa}} : \boldsymbol{\theta}^T \mathbf{x} = 0\}$ such that the retailers (except for i^*) on the same side of the hyperplane as i^* (not inclusive) should not be served, while those on the other side of the hyperplane (not inclusive) should be served. An equivalent feasible solution can then be obtained by adjusting the values of z_i for i^* and the retailers on the hyperplane defined by $\boldsymbol{\theta}$. In short, Proposition 5 finds equivalent feasible solutions of $\mathcal{P}(S, \Psi(\cdot))$ by fixing z_i to 0 or 1 for certain $i \in S$, i.e., we can get problems equivalent to $\mathcal{P}(S, \Psi(\cdot))$ but with fewer decision variables.

Based on this result, we develop Algorithm 2 that recursively reduces the problem $\mathcal{R}(\boldsymbol{\beta})$ to a sequence of problems with $\hat{\kappa}$ decision variables. Consider any two sets $F, U \subseteq I$ satisfying the following conditions:

(C1) The sets $\{\hat{\mathbf{b}}_i : i \in F\}$ and $\{\hat{\mathbf{b}}_i : i \in U\}$ contain $|F|$ and $\hat{\kappa} - |F|$ linearly independent vectors, respectively.

(C2) Any non-zero linear combination of $\{\hat{\mathbf{b}}_i : i \in F\}$ is linearly independent of any non-zero linear combination of $\{\hat{\mathbf{b}}_i : i \in U\}$.

For any $\Psi : \mathbb{R}_+^{\hat{\kappa}} \mapsto \mathbb{R}$, the function `solveP`$(F, U, \Psi(\cdot))$ in Algorithm 2 returns an optimal solution $(z_i^* \, \forall i \in F \cup U)$ and the optimal value v^* of the problem $\mathcal{P}(F \cup U, \Psi(\cdot))$ defined in (8).

For any $i \in F$, the corresponding variable z_i is one of the $\hat{\kappa}$ decision variables in the decomposed problems. For any $i \in U$, the function `solveP`$(F, U, \Psi(\cdot))$ determines whether z_i appears in the decomposed problems. More specifically, the function `solveP`$(F, U, \Psi(\cdot))$ picks a retailer i^* from the set U. In order to reduce the number of decision variables, as suggested by Proposition 5, we

Algorithm 2: Solving $\mathcal{R}(\boldsymbol{\beta})$ by calling the function $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$

1 **function** $\texttt{solveP}(F, U, \Psi(\cdot))$

2 **if** $U = \emptyset$ **then** solve the problem $\mathcal{P}(F, \Psi(\cdot))$, and **return** its optimal solution $(z_i^* \, \forall i \in F)$ and optimal value v^*;

3 let $z_i^* := 0$ for all $i \in F \cup U$, $v^* := \Psi(\mathbf{0})$, and $A := \emptyset$;

4 choose an arbitrary $i^* \in U$;

5 **if** $\hat{\mathbf{b}}_{i^*}$ *can be represented as a linear combination of* $\{\hat{\mathbf{b}}_i : i \in U \setminus \{i^*\}\}$ **then**

6 let $\bar{U} := U \setminus \{i^*\}$, and define $\bar{\Psi}(\mathbf{x}) := \Psi(\hat{\mathbf{b}}_{i^*} + \mathbf{x})$ for any $\mathbf{x} \in \mathbb{R}_+^{\hat{\kappa}}$;

7 obtain the optimal solution $(\bar{z}_i \, \forall i \in F \cup \bar{U})$ and optimal value $\bar{v}$ of the problem $\mathcal{P}(F \cup \bar{U}, \bar{\Psi}(\cdot))$ by calling the function $\texttt{solveP}(F, \bar{U}, \bar{\Psi}(\cdot))$;

8 **if** $\bar{v} < v^*$ **then** let $z_{i^*}^* := 1$, $z_i^* := \bar{z}_i$ for all $i \in F \cup \bar{U}$, and $v^* := \bar{v}$;

9 **end**

10 **foreach** $U_f \subseteq U \setminus \{i^*\}$ *such that* $|U_f| = \hat{\kappa} - |F| - 1$ *and* $\{\hat{\mathbf{b}}_i : i \in U_f \cup \{i^*\}\}$ *are linearly independent* **do**

11 find a vector $\boldsymbol{\theta}$ such that $\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1$ and $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0$ for all $i \in F \cup U_f$;

12 **if** $\boldsymbol{\theta} \in A$ **then** go to Line 17;

13 let $\bar{F} := F \cup \{i^*\}$, $\bar{U}(\boldsymbol{\theta}) := \{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0\}$, and

$$\bar{\Psi}(\boldsymbol{\theta}, \mathbf{x}) := \Psi\left(\mathbf{x} + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0} \hat{\mathbf{b}}_i\right)$$

for any $\mathbf{x} \in \mathbb{R}_+^{\hat{\kappa}}$;

14 **if** $|\bar{U}(\boldsymbol{\theta})| \geq \hat{\kappa} - |F|$ **then** let $A := A \cup \{\boldsymbol{\theta}\}$;

15 obtain an optimal solution $(\bar{z}_i \, \forall i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta}))$ and the optimal value $\bar{v}$ of the problem $\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$ by calling the function $\texttt{solveP}(\bar{F}, \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$;

16 **if** $\bar{v} < v^*$ **then** let $z_i^* := \bar{z}_i$ for all $i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta})$, $z_i^* := 1$ for all $\{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0\}$, $z_i^* := 0$ for all $\{i \in U \setminus \{i^*\} : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i > 0\}$, and $v^* := \bar{v}$;

17 **end**

18 **return** $(z_i^* \, \forall i \in F \cup U)$ and v^* as an optimal solution and the optimal value of the problem $\mathcal{P}(F \cup U, \Psi(\cdot))$, respectively;

19 **end**

can set z_{i^*} to 1 and remove i^* from U. In the meantime, we also enumerate possible partitions of the set $U \setminus \{i^*\}$. Similar to Algorithm 1, it is sufficient to consider the partitions defined by i^*, the retailers in the set F, and $\hat{\kappa} - |F| - 1$ retailers in the set $U \setminus \{i^*\}$ (cf. Line 11 of Algorithm 2). Given such a partition, for any retailer i in the set $U \setminus \{i^*\}$ that is not on the partition, the value of z_i can be fixed according to Proposition 5. Then i^* is added to the set F and the set U is updated to the retailers in $U \setminus \{i^*\}$ on the partitioning hyperplane. By calling $\texttt{solveP}(F, U, \Psi(\cdot))$ recursively, we eventually obtain $\hat{\kappa}$ retailers in F and reduce U to an empty set. In this case, the $\hat{\kappa}$ retailers in F yields a $\hat{\kappa}$-variable problem in the form of $\mathcal{P}(S, \Psi(\cdot))$.

Proposition 6. *The function* $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$ *defined in Algorithm 2 solves* $\mathcal{R}(\boldsymbol{\beta})$ *for any* $\boldsymbol{\beta}$ *constructed in Line 3 of Algorithm 1, by decomposing it into at most* $C^{\hat{\kappa}}_{|H(\boldsymbol{\beta})|}$ *problems in the form of* $\mathcal{P}(S, \Psi(\cdot))$ *with* $\hat{\kappa}$ *decision variables.*

We also note that $\mathcal{R}(\boldsymbol{\beta})$ can be solved by enumerating all possible subsets of $H(\boldsymbol{\beta})$ whose corresponding values of z_i could be fractional. According to Theorem 2, as an optimal solution has at most $\hat{\kappa}$ fractional components, $H(\boldsymbol{\beta})$ has $C^{\hat{\kappa}}_{|H(\boldsymbol{\beta})|}$ subsets to be potentially fractionally served, each of which has $\hat{\kappa}$ retailers. Given such a subset S, whether to serve a retailer in $H(\boldsymbol{\beta}) \setminus S$ is unknown. As their corresponding values of z_i can only be 0 or 1, the best we can do is to enumerate all possible 0-1 assignments for the $|H(\boldsymbol{\beta})| - \hat{\kappa}$ retailers, which yields $2^{|H(\boldsymbol{\beta})| - \hat{\kappa}}$ combinations. Based on the binary assignment for retailers in $H(\boldsymbol{\beta}) \setminus S$, we can then determine the values of z_i for the retailers in S by solving a problem in the form of $\mathcal{P}(S, \Psi(\cdot))$. As a result, if $\mathcal{R}(\boldsymbol{\beta})$ is to be solved by enumeration, then $C^{\hat{\kappa}}_{|H(\boldsymbol{\beta})|} \times 2^{|H(\boldsymbol{\beta})| - \hat{\kappa}}$ $\hat{\kappa}$-variable problems in the form of $\mathcal{P}(S, \Psi(\cdot))$ must be solved. In contrast, Proposition 6 shows that Algorithm 2 only needs to solve $C^{\hat{\kappa}}_{|H(\boldsymbol{\beta})|}$ such problems, which clearly has a better performance. This comparison also suggests that Algorithm 2 may still enumerate all subsets of $H(\boldsymbol{\beta})$ with $\hat{\kappa}$ retailers. However, by choosing the right order to enumerate these subsets, we can apply Proposition 5 to decide whether to serve a retailer not contained in the fractionally served subset. This eliminates the enumeration of the 0-1 assignments for integrally served retailers and leads to Algorithm 2 that only solves $C^{\hat{\kappa}}_{|H(\boldsymbol{\beta})|}$ problems with $\hat{\kappa}$ decision variables.

3.4 Solving the Relaxation of $\mathcal{Q}$

According to Propositions 4 and 6, the relaxation of $\mathcal{Q}$ can be solved by Algorithm 1, which calls the function $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$ in Line 11 to solve the problem $\mathcal{R}(\boldsymbol{\beta})$. Through this approach, the relaxation of $\mathcal{Q}$ is decomposed into at most $C^{\hat{\kappa}}_n$ problems with $\hat{\kappa}$ decision variables.

Theorem 3. *The relaxation of $\mathcal{Q}$ can be decomposed into at most $C_n^{\hat{\kappa}}$ problems in the form of $\mathcal{P}(S,\Psi(\cdot))$ with $\hat{\kappa}$ decision variables.*

Note that $C_n^{\hat{\kappa}}$ is in the order of $O(n^{\hat{\kappa}})$, which is polynomial in the number of retailers n. For most applications, $\hat{\kappa} \ll n$ is a small integer. Based on the KKT conditions, optimizing a $\hat{\kappa}$-variable $\mathcal{P}(S,\Psi(\cdot))$ can often be further reduced to subproblems whose solutions are either obtained in a closed form or computed by standard numerical methods for solving equation systems, e.g., the algorithms used by the commercial software package Mathematica. For illustration, we show in Appendix C of the Online Supplement how to solve the decomposed problems with $\hat{\kappa}$ variables for the two applications considered in Section 4 for numerical study. Furthermore, when $\Phi(\cdot)$ is quasi-concave, Proposition 2 shows that there exists an optimal solution with at most one fractional component. Thus, each $\hat{\kappa}$-variable problem can be further decomposed into $\hat{\kappa} \times 2^{\hat{\kappa}-1}$ single-variable problems. As a result, the relaxation of $\mathcal{Q}$ can be solved rapidly for small $\hat{\kappa}$.

Observe that Algorithms 1 and 2 enumerate potential partitioning hyperplanes in Lines 2 and 10, respectively. This idea has been exploited in the joint location-inventory literature to solve the subproblems obtained from Lagrangian relaxation or column generation. These subproblems can be viewed as special cases of $\mathcal{Q}$ where $\Phi(\cdot)$ is a univariate concave function with $\kappa = 1$ (Daskin et al., 2002 and Shen et al., 2003), a sum of two such functions with $\kappa = 2$ (Shu et al., 2005), or a sum of three such functions with $\kappa = 3$ (Shen and Qi, 2007). In their algorithms, the vectors $\mathbf{b}_i/a_i$ for all $i \in I$ are sorted before starting the enumeration of hyperplanes so as to speed up the evaluation of the objective function value corresponding to each hyperplane. This pre-sorting technique can also be adopted in Algorithms 1 and 2 to sort $\mathbf{b}_i/a_i$ and $\hat{\mathbf{b}}_i$, respectively. It is capable to reduce the computation of $\sum_{i\in S_1(\boldsymbol{\beta})} \mathbf{b}_i$ in Line 11 of Algorithm 1 to $O(\kappa)$, that of $\sum_{i\in S_1(\boldsymbol{\beta})} a_i$ in Line 12 of Algorithm 1 to $O(1)$, and that of $\sum_{i\in U:\boldsymbol{\theta}^T\hat{\mathbf{b}}_i<0} \hat{\mathbf{b}}_i$ in Line 13 of Algorithm 2 to $O(\hat{\kappa})$. This would significantly improve the performance of Algorithms 1 and 2 when the $\hat{\kappa}$-variable problems in the form of $\mathcal{P}(S,\Psi(\cdot))$ can be solved very efficiently.

As the concluding remark of this section, we would like to compare the proposed approach to solve the relaxation of $\mathcal{Q}$ (and $\mathcal{Q}$ itself) with the existing approaches for some special cases. In many cases, with the aid of the pre-sorting technique, our approach is as efficient as the best known algorithm. For example, the relaxation of $\mathcal{Q}$ for the capacitated facility location problem in Application 1 can be written as

$$\min \left\{ -\sum_{i=1}^{n} a_i z_i \;\middle|\; \sum_{i=1}^{n} b_i^1 z_i \le C,\; z_i \in [0,1]\ \forall i \in I \right\}.$$

Solving the above LP requires sorting a_i/b_i^1 for all i and the computational complexity is $O(n\log n)$. In the meantime, if we adopt Algorithm 1 and pre-sort b_i^1/a_i before the enumeration in Line 2, it is easy to see that computational complexity of Algorithm 1 is dominated by sorting b_i^1/a_i and hence it is $O(n\log n)$ as well.

This observation also applies to the subproblems arising from the uncapacitated location-inventory models. As $\Phi(\cdot)$ is concave in these models, $\mathcal{Q}$ and its relaxation are equivalent. Furthermore, the $\hat{\kappa}$-variable problem in the form of $\mathcal{P}(S,\Psi(\cdot))$ must have an optimal solution in $\{0,1\}^{\hat{\kappa}}$, so it can be solved by evaluating the optimal values of the $2^{\hat{\kappa}}$ solutions in $\{0,1\}^{\hat{\kappa}}$. If we pre-sort the vectors $\mathbf{b}_i/a_i$ as in Daskin et al. (2002), Shen et al. (2003), Shu et al. (2005), and Shen and Qi (2007), Algorithm 1 essentially reduces to the corresponding algorithms in the aforementioned literature that solves the instances of $\mathcal{Q}$ with concave objective functions and $\kappa = 1, 2, 3$, respectively.

For the capacitated location-inventory problem studied in Ozsen et al. (2008), the subproblem yielded by Lagrangian relaxation is in the form of

$$\min\left\{-\sum_{i=1}^{n} a_i z_i + \Phi\left(\sum_{i=1}^{n} b_i^1 z_i\right) \,\middle|\, \sum_{i=1}^{n} b_i^1 z_i \le C,\ z_i \in \{0,1\}\ \forall i \in I\right\}, \tag{9}$$

where $\Phi(\cdot)$ is a nondecreasing function. We note that the original constraint considered in Ozsen et al. (2008) is $f(\sum_{i=1}^n b_i^1 z_i) \le \tilde{C}$ for an increasing function $f(\cdot)$, which, due to the monotonicity of $f(\cdot)$, is equivalent to $\sum_{i=1}^n b_i^1 z_i \le C$ for some C. They first consider the relaxation of (9), which allows $z_i \in [0,1]$ for all $i \in I$, and then obtain an integral solution to (9) by branch-and-bound. Again, by pre-sorting b_i^1/a_i, Algorithm 1 reduces to the approach adopted by Ozsen et al. (2008) to solve the relaxation of (9). In other words, our proposed approach to solve (9) is equivalent to that in Ozsen et al. (2008).

In the meantime, we acknowledge that Algorithm 1, targeting to solve a very generic problem without assuming any problem-specific property, may fail to be the most efficient algorithm for some special cases. For example, Desrochers et al. (1995) consider the column generation algorithm for the congested multi-sourcing facility location problem. The corresponding pricing problem is an instance of the relaxation of $\mathcal{Q}$ with convex $\Phi(\cdot)$ and $\kappa = 1$, which is a convex optimization problem. Finding its optimal solution could require Algorithm 1 to solve n univariate problems in the form of $\mathcal{P}(S,\Psi(\cdot))$. Desrochers et al. (1995) utilize the property that a local optimum to a convex optimization problem must be a global optimum and propose an algorithm for the pricing problem based on Fibonacci's search, which reduces the times of solving the univariate $\mathcal{P}(S,\Psi(\cdot))$ problems.

As a result, the approach by Desrochers et al. (1995) should be more efficient than Algorithm 1 if the univariate problems are hard to solve. However, if the univariate problems can be solved in $O(1)$, e.g., when $\Phi(\cdot)$ is a quadratic function, Algorithm 1 would perform as well as the approach by Desrochers et al. (1995). In this case, the computational complexity of Algorithm 1 with pre-sorting b_i^1/a_i is $O(n \log n)$, whereas the complexity of the approach by Desrochers et al. (1995) is at least $O(n \log n)$ because it requires sorting a_i. We also note that it is difficult to generalize the approach by Desrochers et al. (1995) to the instances with convex $\Phi(\cdot)$ and $\kappa > 1$.

4 Computational Study

In this section, we implement the facility location and production/capacity planning models with economies and diseconomies of scale, which, respectively, correspond to Applications 3 and 4 in Section 2, to establish the effectiveness of the proposed solution approach, understand the managerial implications, and demonstrate the importance of incorporating the general facility costs. The algorithm is coded in C++. All experiments are conducted on a Dell desktop with 3.20 GHz Intel i7 CPU and 16 GB memory running the Windows operating system (64-bit). The data sets are provided in the online supplements.

Throughout the computational study, the initial columns at the root node of the branch-and-price tree consist of the columns of $(j, \emptyset)$ for all $j \in J$ and those corresponding to an optimal solution to the problem that only considers the linear cost components in the instance. At any other node of the tree, the initial columns are obtained from the final master problem of its parent node. All the master problems are solved by the CPLEX Academic Initiative Edition 12.5 (64-bit) solver in its default environment. When solving the pricing problem, we first employ a heuristic inspired by Algorithm 1, which enumerates the partitioning hyperplanes as in Algorithm 1. For each hyperplane containing a small number (e.g., $\hat{\kappa}$) of retailers, it assigns 0-1 values to the retailers not on the hyperplane according to Algorithm 1 and enumerates all possible binary assignments for the retailers on the hyperplane. The branch-and-bound procedure for the pricing problem (see Algorithm 3 in Appendix B of the Online Supplement) is applied only when the heuristic fails to find a column with a negative reduced cost. The branch-and-bound procedure in Algorithm 3 is also terminated as long as an integral solution with a negative reduced cost is identified. Numerical experiments verify that this column generation strategy is superior to the one that always solves the pricing problem to optimality.

4.1 Facility Location and Production

We first consider the facility location and production problem with economies and diseconomies of scale, which corresponds to Application 3. Following the notation in Application 3, b_i^k for all $k \in \{1, ..., \kappa\}$ denote the constant demand rates of product k at retailer i, respectively. If a facility at site j is set up, the production cost for facility j to serve the retailers in the set S is defined as $\Gamma_j(\sum_{i\in S} b_i^1, ..., \sum_{i\in S} b_i^\kappa) = \sum_{k=1}^{\kappa} \Gamma_j^k(\sum_{i\in S} b_i^k)$. Moreover, for any $k \in \{1, ..., \kappa\}$, let c_{ij}^k be the transportation cost to send one unit of product k from facility j to retailer i. Then the linear transportation cost from facility j to retailer i is $t_{ij} = \sum_{k=1}^{\kappa} c_{ij}^k b_i^k$.

For $\kappa \in \{2, 3, 4\}$, we test on the random instances with $|J|, |I| \in \{20, 40, 60, 80, 100\}$, which denote the numbers of the potential facility locations and the retailers, respectively. Given a combination of κ, $|J|$, and $|I|$, 20 instances are generated randomly by the following procedure. The locations of the potential facilities and the retailers are uniformly distributed over $[0, 10] \times [0, 10]$. For any retailer i and product k, the demand rate b_i^k is randomly generated in $[10, 50]$. The transportation cost parameters, i.e., c_{ij}^k for all $k \in \{1, ..., \kappa\}$, are set to the Euclidean distance between i and j times a factor for each product that is generated uniformly in $[50, 100]$. The fixed location cost F_j for any facility j is randomly generated in $[500, 1500]$.

To model the economies and diseconomies of scale in production, the production costs are set to inverse S-shaped functions of the production quantities. In particular, given the production quantity q_j^k of product k at facility j, the production cost is $\Gamma_j^k(q_j^k) = 10^{-3}(q_j^k - e_j^k)^3 + 10^{-3}(e_j^k)^3$ with e_j^k uniformly generated in $[100, 500]$. Obviously, $\Gamma_j^k(q_j^k)$ is strictly increasing in q_j^k. It is a concave function when $q_j^k \le e_j^k$ and a convex function when $q_j^k \ge e_j^k$. In other words, economies of scale exist when the production quantity is less than e_j^k, whereas diseconomies of scale take place when the production quantity exceeds e_j^k. Consequently, e_j^k is referred to as the economic point. Figure 1 illustrates the inverse S-shaped production cost functions under different economic points. Note that the production cost functions are the same as the cubic functions used in Lu et al. (2014) except that the function values are enlarged by 10 times. Furthermore, this implementation example can be viewed as a variation of the model in Lu et al. (2014) in a κ-product single-sourcing setting. Although Lu et al. (2014) consider a capacitated model, we note that the capacitated version is no harder than the uncapacitated one used in our experiments.

Tables 1, 2, and 3 demonstrate the computational effectiveness of our approach applied to the facility location and production problem with $\kappa = 2, 3, 4$, respectively. The first two columns "$|J|$" and "$|I|$" show the input size of each instance class, i.e., the numbers of potential facility locations

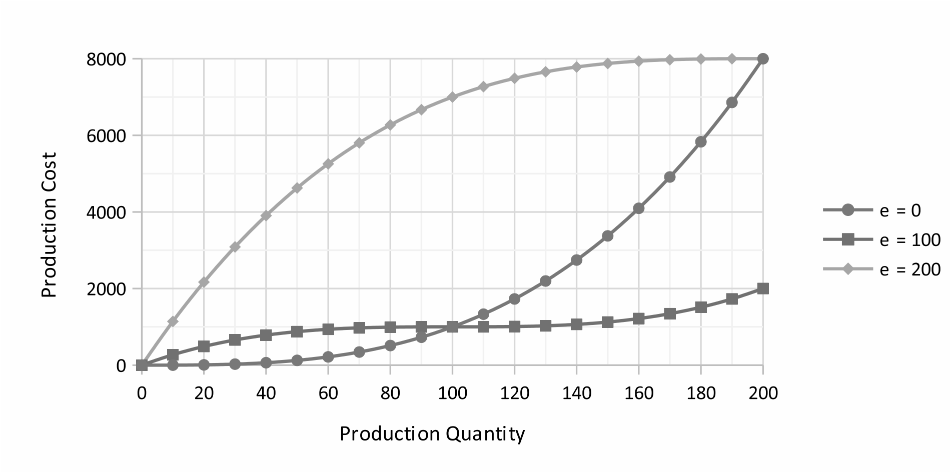

Figure 1: Inverse S-Shaped Production Costs

and retailers, respectively. The columns "B&P nodes," "Iter.," "Col.," "B&B nodes," and "Sub-prob." report the averages of nodes in the branch-and-price trees, times that the master problems are solved, columns added, nodes explored by the B&B procedure for solving the pricing problems, and κ-variable decomposed subproblems in the form of $\mathcal{P}(S, \Psi(\cdot))$ solved for solving the pricing problems, respectively. The columns titled "Time" display the minimal, average, and maximal CPU times in seconds taken to solve the randomly generated instances of each input size. The column "Fac. open" gives the average number of facilities open in the optimal binary solutions obtained by the branch-and-price algorithm. The last column "Cost" reports the average of the corresponding optimal values in thousands, respectively.

As shown in Table 1, when $\kappa = 2$, the proposed branch-and-price algorithm can be applied to solve instances with up to 100 potential facilities and 100 retailers within moderate CPU times. The average CPU times are always bounded by 15 minutes. Furthermore, the solution difficulty increases with the number of retailers, but it is rather insensitive to the number of potential facilities.

Theorem 2 shows that the number of fractional components in an optimal solution of the relaxation of the pricing problem increases in κ, which should make it harder for the B&B algorithm to find an integral solution ("B&B nodes"). According to Theorem 3, to solve the relaxation of the pricing problem, we need to solve a series of κ-variable decomposed subproblems in the form of $\mathcal{P}(S, \Psi(\cdot))$, whose number ("Sub-prob.") may grow exponentially in κ. Consequently, the pricing problem would get much more difficult as κ increases. Nevertheless, the proposed algorithm is still capable of solving practical-scale instances with up to 80 and 60 potential facilities and retailers for κ equal to 3 and 4, respectively. Also note that the decisions of the facility location and production

Table 1: Algorithmic Performance for Facility Location and Production Problems with $\kappa = 2$

$\|J\|$	$\|I\|$	B&P nodes	Iter.	Col.	B&B nodes	Sub-prob.	Time			Fac. open	Cost ($\times 10^3$)
							Min	Avg	Max		
20	20	1	11	2871	24	28	0	0	1	5	257.54
20	40	7	63	19615	146	2743	1	4	25	6	480.72
20	60	16	210	58004	320	33424	6	47	356	7	705.29
20	80	20	359	96674	402	51721	38	154	479	9	892.35
20	100	79	1200	188361	1578	312198	58	864	2957	10	1115.07
40	20	1	9	2426	40	8	0	0	1	6	228.20
40	40	2	34	17038	96	1213	1	3	12	8	422.15
40	60	22	161	54354	860	24149	6	44	305	9	597.96
40	80	45	448	106505	1812	113485	27	294	2075	10	734.35
40	100	24	356	154919	976	82907	86	428	1178	12	919.15
60	20	1	8	2271	66	0	0	0	1	7	211.85
60	40	2	24	13771	102	245	1	2	4	9	374.55
60	60	9	100	45473	552	7590	7	24	102	10	538.12
60	80	17	190	90419	996	53910	27	133	903	12	679.61
60	100	35	397	149850	2124	120885	67	486	1740	14	827.59
80	20	1	8	2118	80	0	0	1	1	8	197.57
80	40	3	28	14196	248	328	1	3	6	9	356.80
80	60	6	61	41818	440	4312	7	19	41	11	510.34
80	80	18	140	78767	1416	27796	27	98	586	13	637.03
80	100	54	438	139363	4344	160839	84	534	1812	14	788.40
100	20	1	7	1971	100	0	0	1	1	8	191.09
100	40	2	24	12585	240	162	1	3	7	10	339.86
100	60	5	56	38601	480	3243	12	20	56	12	484.56
100	80	13	121	76283	1260	30784	18	90	513	14	614.35
100	100	24	219	123516	2440	81051	85	270	1726	16	753.12

Table 2: Algorithmic Performance for Facility Location and Production Problems with $\kappa = 3$

$\lvert J \rvert$	$\lvert I \rvert$	B&P nodes	Iter.	Col.	B&B nodes	Sub-prob.	Time			Fac. open	Cost ($\times 10^3$)
							Min	Avg	Max		
20	20	1	11	3203	20	180	0	7	40	5	441.41
20	40	6	74	27065	124	18086	15	760	7320	6	794.58
20	60	5	104	73928	94	52709	213	2463	14830	7	1124.45
20	80	13	212	140566	256	209968	878	11130	51895	8	1443.16
40	20	1	9	2474	44	8	0	1	7	6	372.78
40	40	2	32	18992	76	3040	4	155	1081	7	669.94
40	60	7	93	61465	284	45026	179	2446	15917	8	948.74
40	80	9	134	114595	368	115875	290	6124	36582	10	1229.35
60	20	1	8	2042	60	0	0	0	1	7	345.65
60	40	4	33	16134	228	4888	4	247	1434	9	629.45
60	60	5	68	50311	276	17806	192	1026	2928	9	890.89
60	80	12	126	99654	702	126248	626	6906	70968	11	1100.84
80	20	1	7	1850	80	0	0	0	1	7	329.97
80	40	1	21	13271	96	926	4	55	259	9	596.42
80	60	5	57	44023	368	12823	117	772	3330	10	838.13
80	80	5	91	91370	408	54431	611	3339	10594	12	1063.27

Table 3: Algorithmic Performance for Facility Location and Production Problems with $\kappa = 4$

$\|J\|$	$\|I\|$	B&P nodes	Iter.	Col.	B&B nodes	Sub-prob.	Time Min	Time Avg	Time Max	Fac. open	Cost ($\times 10^3$)
20	20	1	12	3685	20	283	0	174	906	4	608.96
20	40	2	46	29529	46	19493	831	14736	60493	6	1085.70
20	60	1	77	82617	28	62311	13035	45901	153500	7	1516.16
40	20	1	9	2554	44	66	0	41	409	6	533.88
40	40	2	36	21972	92	13355	321	10167	47632	7	977.29
40	60	2	64	67023	64	47734	7717	36529	101717	8	1369.79
60	20	1	8	2118	78	116	0	73	521	6	511.46
60	40	2	28	17835	114	7163	324	5411	25500	8	932.25
60	60	2	58	61050	96	38531	3116	27139	68448	9	1277.42

problem, i.e., location and retailer-facility assignment, have a very long-term impact and are made very infrequently. Although the average computational time could be 1.9 hours for instances with $\kappa = 3$ and $|I| = 80$ and 10.1 hours for instances with $\kappa = 4$ and $|I| = 60$, the proposed algorithm remains practical as it is very likely that such a strategic decision problem will only be solved once in every few years.

We also observe from Tables 1, 2, and 3 that compared with the number of columns generated ("Col."), the number of column generation iterations ("Iter.") is rather small. This is because we add almost all columns identified with a negative reduced cost to the master problem in each column generation iteration. This strategy is advantageous since the LP master problems can be solved very efficiently, while solving the pricing problems could be potentially time consuming. Moreover, as aforementioned, we implement a heuristic to search for an integral solution for the pricing problem and terminate the B&B algorithm whenever a column with a negative reduced cost is identified. Consequently, only a small number of B&B nodes are enumerated when solving the pricing problems ("B&B nodes"). Also note that when solving the pricing problem to optimality, we choose to enumerate all possible integral solutions when the number of decision variables in the pricing problem (or the problem at a B&B node) is small (see Line 5 of Algorithm 3 in Appendix B of the Online Supplement). This explains why we solve a very small number of κ-variable decomposed subproblems in the form of $\mathcal{P}(S, \Psi(\cdot))$ ("Sub-prob.") when $|I| = 20$. There are also several instance classes where no such problems are solved at all.

As expected, Tables 1, 2, and 3 show that the optimal cost ("Cost") increases in the number of products κ and the number of retailers $|I|$, while it decreases in the number of potential facilities $|J|$. The number of open facilities ("Fac. open") then increases in both $|I|$ and $|J|$. It is also noted that the number of open facilities is insensitive to κ. This is mainly because at any facility, the total facility cost is defined as the sum of the facility costs for individual products.

In order to investigate the impact of the economic points on the algorithmic performance as well as the model solution, we consider the case with two products, i.e., $\kappa = 2$. Two basic instances with $|J| = |I| = 40$ and $|J| = |I| = 100$ are generated by uniformly choosing b_i^1 and b_i^2 for all i in $[10, 20]$ and $[20, 50]$, respectively. Variant instances are then obtained by setting the economic points at any potential facility, i.e., (e_j^1, e_j^2) for all $j \in J$, to a specific combination chosen from $e_j^1, e_j^2 \in \{0, 50, 100, 200, 500\}$. In addition, all the facility candidates have a fixed location cost of 10000 so as to eliminate the influence of the location costs on the location decisions. The results are summarized in Table 4. The columns "e_j^1" and "e_j^2" show the different economic point combinations for the two products at all potential facilities. The columns "Time," "Fac. open," and "Cost" display the computational time to solve the instance, the optimal number of facilities open, and the optimal value. For the purpose of comparison, we also solve the corresponding uncapacitated facility location problem (UFLP) of each instance, i.e., the problem without the terms Γ_j in model $\mathcal{P}$. Note that the economic points have no impact on the optimal UFLP solution, which has 10 and 17 facilities open for all instances with $|J| = |I| = 40$ and $|J| = |I| = 100$, respectively. As the optimal UFLP solution is also feasible to model $\mathcal{P}$, we can evaluate its corresponding objective value of model $\mathcal{P}$. The columns "Reduct." present how much cost can be saved in percentage if we implement our solution to model $\mathcal{P}$ instead of the optimal UFLP solution. We also compare the facilities open in the solutions of model $\mathcal{P}$ and the UFLP, and report in the columns "Open in both" the numbers of facilities open in both solutions.

Based on Table 4, we first examine the impact of the economic points on the number of facilities open in model $\mathcal{P}$.

Observation 1. *The number of facilities open is generally nonincreasing with the increase of the economic points.*

This observation results from the trade-off between economies-of-scale and diseconomies-of-scale in the inverse S-shaped facility cost functions. When the economic points are close to zero, the

Table 4: Impact of Economic Points

		$\|J\| = \|I\| = 40$, 10 facilities open in UFLP					$\|J\| = \|I\| = 100$, 17 facilities open in UFLP				
e_j^1	e_j^2	Time	Fac. open	Cost ($\times 10^3$)	Reduct. (%)	Open in both	Time	Fac. open	Cost ($\times 10^3$)	Reduct. (%)	Open in both
0	0	3	13	295.99	4.42	10	190	26	588.81	18.94	11
0	50	1	11	277.32	2.52	9	64	24	535.65	13.97	12
0	100	2	10	279.59	0.55	9	58	20	526.78	7.94	13
0	200	1	8	335.18	1.43	5	590	15	621.34	1.56	14
0	500	872	2	759.23	27.10	0	15544	6	1490.84	28.58	2
50	0	4	12	295.97	3.93	9	187	25	586.25	18.07	10
50	50	1	11	276.78	2.16	9	156	24	531.93	13.06	12
50	100	2	10	278.36	0.43	9	59	20	519.79	7.41	13
50	200	2	8	332.46	1.77	5	423	14	608.01	1.99	10
50	500	679	2	737.66	29.06	0	9138	6	1437.65	30.77	2
100	0	2	11	304.42	3.44	9	178	24	605.27	16.75	12
100	50	1	11	285.00	1.74	9	115	23	550.26	11.71	13
100	100	1	10	285.88	0.29	9	51	20	535.08	6.59	13
100	200	1	8	338.48	2.07	4	547	14	615.84	2.53	10
100	500	342	2	724.80	30.77	0	6974	6	1407.56	32.59	2
200	0	2	11	347.12	2.45	9	148	24	709.37	13.15	13
200	50	2	11	327.70	0.89	9	62	21	648.61	9.04	13
200	100	4	10	327.11	0.06	9	462	20	632.52	4.54	13
200	200	3	7	375.74	2.71	3	1063	13	696.08	3.54	8
200	500	32	2	725.34	33.31	0	9378	5	1409.28	35.29	2
500	0	4	10	684.64	0.46	9	304	21	1544.92	4.70	13
500	50	4	9	661.35	0.19	9	698	20	1473.71	2.88	12
500	100	5	8	653.08	0.94	8	1806	15	1437.45	2.01	10
500	200	10	6	686.31	4.44	4	4360	11	1437.28	5.81	5
500	500	19	2	936.96	33.99	0	5931	5	1910.63	35.93	2

effect of diseconomies-of-scale dominates, due to which demand pooling is less cost efficient and the optimal network design solution tends to open more facilities. In contrast, with the increase of the economic points, the effect of economies-of-scale starts to take over gradually, which leads to network consolidation. Furthermore, the phenomenon is much more significant for the economic point of product 2, because the demand rates of product 2 generated in $[20, 50]$ is much higher than that of product 1 generated in $[10, 20]$.

We also observe how the supply chain cost changes with respect to the economic points.

Observation 2. *The supply chain cost first decreases and then increases with the increase of the economic points.*

For any given q_j^k, the first derivative of the inverse S-shaped facility cost function $\Gamma_j^k(q_j^k)$ with respect to e_j^k shows that $\Gamma_j^k(q_j^k)$ decreases in e_j^k if $2e_j^k < q_j^k$ and increases in e_j^k if $2e_j^k > q_j^k$. When the economic point e_j^k is low, e.g., $e_j^k = 0$, the production quantity q_j^k at any of the open facilities is much higher than $2e_j^k$. In this case, if we keep the supply chain configuration and slightly increase e_j^k, the total supply chain cost decreases as the production cost $\Gamma_j(q_j^k)$ decreases in e_j^k. Under the increased economic point, the supply chain cost can be further reduced by optimizing the supply chain configuration. Similarly, if e_j^k is high, e.g., $e_j^k = 500$, it is very unlikely for the production quantity q_j^k at an open facility to exceed $2e_j^k$, which implies that the production cost $\Gamma_j(q_j^k)$ increases in e_j^k. Therefore, when we decrease e_j^k, the supply chain cost is reduced under the same network configuration, and can be further improved by solving model $\mathcal{P}$ using the lowered e_j^k. As a result, we obtain Observation 2.

Table 4 shows that most of the instances are solved in less than 20 minutes. For the three instances requiring more than 2 hours to solve, i.e., the instances with $|I| = |J| = 100$, $e_j^1 \in \{0, 50, 200\}$, and $e_j^2 = 500$, the long CPU time is mainly caused by the large number of branch-and-price nodes explored to obtain an integral solution, which are 311, 197, and 83, respectively. Moreover, it reveals that the computational time shares the same trend as the supply chain cost under varying economic points.

Observation 3. *Generally speaking, the solution difficulty first decreases and then increases with the increase of the economic points.*

The increase of the economic points makes the facility cost functions dominated by the concave segments, which, intuitively, should reduce the solution difficulty of the problem. However, as shown in the explanation of Observation 2, the production cost at each open facility first decreases and then increases in the economic points. When the production costs are low, model $\mathcal{P}$ focuses on decreasing the location and transportation costs. When the production costs are high, it is more important to reduce the production costs, which, due to the nonlinearity of the production cost functions, is more difficult than cutting the location and transportation costs. Consequently, we observe shorter CPU times for moderate economic points.

Next, we compare the solutions of model $\mathcal{P}$ and the UFLP. In 94% of all instances, the number of facilities opened by both model $\mathcal{P}$ and the UFLP, which is shown in the columns "Open in both," is strictly less than the minimum of the numbers of facilities opened by these two models, respectively. We obtain the following implication.

Observation 4. *If model $\mathcal{P}$ (resp. the UFLP) opens no more facilities than the UFLP (resp. model $\mathcal{P}$), the set of facilities opened by model $\mathcal{P}$ (resp. the UFLP) may not be a subset of that by the UFLP (resp. model $\mathcal{P}$).*

Observation 4 can be interpreted by the property that the optimal set of facilities open in a p-median problem may not be a subset of that in the corresponding p'-median problem, where $p < p'$. The observation implies that the optimal solution of model $\mathcal{P}$ cannot be easily obtained from that of the UFLP counterpart, which demonstrates the necessity of solving model $\mathcal{P}$ efficiently. One may notice that when the number of open facilities in model $\mathcal{P}$ is very close to that in the UFLP, the two solutions may share a large set of open facilities. However, as the number of open facilities cannot be predicted without solving model $\mathcal{P}$, it is still important to obtain the solution of model $\mathcal{P}$.

The ultimate test of the necessity to study model $\mathcal{P}$ is how much cost it saves. Table 4 shows that model $\mathcal{P}$ reduces the supply chain cost by 10.41% on average and the reduction can easily exceed 25% when $e_j^2 = 500$. Therefore, we obtain the following observation indicating that model $\mathcal{P}$ can yield significant cost savings.

Observation 5. *Compared with the UFLP, model $\mathcal{P}$ leads to a solution that significantly reduces the systemwide cost of location, transportation, and facility operation.*

4.2 Facility Location and Capacity Planning

Here we use a single-product location and capacity planning model, i.e., Application 4 with $m = 1$ and $\kappa = 2$, to implement model $\mathcal{P}$. The notation is adopted from Application 4 except that the superscript k is dropped for simplicity as we only consider one product. The random test instances are generated as follows. The locations of the potential facilities and the retailers are uniformly distributed over $[0, 10] \times [0, 10]$. For any potential facility j, the fixed location cost F_j is randomly generated in $[1000, 3000]$. Similar to Section 4.1, the capacity cost is modeled as $\Gamma_j(q_j) = 10^{-3}(q_j - e_j)^3 + 10^{-3}e_j^3$, where e_j is the economic point uniformly generated in $[100, 150]$. Moreover, we assume the demand rate follows an independent normal distribution for any retailer i. The mean demand rate a_i and variance b_i are randomly generated in $[10, 20]$ and $[5, 10]$, respectively. We choose $\epsilon = 0.025$, which corresponds to a 97.5% service level, and so $F^{-1}(1 - \epsilon) = 1.96$. The linear transportation cost t_{ij} from facility j to retailer i is set to the Euclidean distance between i and j multiplied by the expected demand rate a_i and a parameter uniformly generated in [50,100].

Similar to Table 1, we consider 25 test classes by setting the numbers of potential facilities and retailers, i.e., $|J|$ and $|I|$, to any combination of $|J|, |I| \in \{20, 40, 60, 80, 100\}$ and generate 20 random instances for each class. The computational results are summarized in Table 5, which resemble those in Table 1. In particular, the proposed branch-and-price approach can solve all the instances within moderate CPU times. There is only one instance class, i.e., the one with $|J| = 20$ and $|I| = 100$, that requires an average CPU time slightly more than half an hour. All other instance classes can be solved in less than 10 minutes on average.

We next study the impact of the service level ϵ and the economic point e_j on the algorithmic performance and the model output. As in Table 4, two basic instances with $|J| = |I| = 40$ and $|J| = |I| = 100$, respectively, are generated in the same manner as those in Table 5 except that the fixed location costs of all potential facilities are set to 1500 and the demand variance b_i are drawn uniformly in the interval $[50, 100]$. Note that b_i is enlarged to amplify the impact of the service level ϵ. The variant instances are then obtained by considering any $\epsilon \in \{0.2, 0.1, 0.025, 0.005\}$ and setting e_j for all j to any value in the set $\{0, 50, 100, 200, 300\}$. We also consider the solutions optimal to the UFLP counterparts, and compare the corresponding costs and open facilities with those obtained by solving model $\mathcal{P}$ with the nonlinear capacity cost Γ_j. The results are displayed in Table 6.

First, we have Observations 1 and 2 for Table 6, which reveal how the number of facilities open and the supply chain cost change with respect to the economic point e_j, respectively. Moreover, both the number of facilities open and the supply chain cost are nonincreasing with the increase of ϵ. This observation is rather straightforward. Since $1 - \epsilon$ represents the service level of the supply chain, we need more capacity as ϵ decreases, which leads to more open facilities and higher supply chain cost. In addition, all instances in Table 6 are solved within reasonable computational time.

Table 5: Algorithmic Performance for Facility Location and Capacity Planning Problems

$\|J\|$	$\|I\|$	B&P nodes	Iter.	Col.	B&B nodes	Sub-prob.	Time Min	Time Avg	Time Max	Fac. open	Cost ($\times 10^3$)
20	20	1	8	1210	24	0	0	0	1	6	50.20
20	40	1	19	10997	22	26	0	7	38	8	86.76
20	60	3	48	36113	60	381	2	106	625	9	121.78
20	80	2	65	65439	42	719	18	209	719	11	148.52
20	100	8	147	108709	168	7015	115	2022	8266	12	178.35
40	20	1	8	1177	52	0	0	0	1	6	45.49
40	40	2	18	8798	60	36	0	10	66	9	77.83
40	60	2	36	32352	60	104	2	32	315	11	108.27
40	80	3	60	63835	120	597	4	172	640	13	132.30
40	100	4	95	127755	168	1596	71	505	1727	15	157.73
60	20	1	7	1148	78	0	0	0	1	7	42.82
60	40	1	14	8095	72	0	0	1	2	10	73.22
60	60	2	34	29694	114	67	2	22	141	12	100.95
60	80	3	55	61851	192	452	6	135	659	14	126.21
60	100	5	96	135420	276	1305	20	431	1456	16	149.74
80	20	2	7	1188	152	0	0	0	1	7	41.47
80	40	1	14	7171	96	0	1	1	1	10	70.92
80	60	2	35	27920	192	120	2	37	262	12	96.91
80	80	2	50	62402	152	279	8	89	493	14	121.68
80	100	4	85	128438	288	969	20	334	2905	16	143.64
100	20	2	7	1108	160	0	0	0	1	7	40.25
100	40	2	17	7608	230	0	0	2	5	11	69.22
100	60	3	32	26030	270	43	3	17	154	13	95.00
100	80	2	45	57668	170	189	7	64	586	15	117.94
100	100	3	79	126863	280	565	36	220	1273	17	140.69

Table 6: Impact of Service Level and Economic Point

ϵ	e_j	$\|J\| = \|I\| = 40$, 6 facilities open in UFLP					$\|J\| = \|I\| = 100$, 7 facilities open in UFLP				
		Time	Fac. open	Cost ($\times 10^3$)	Reduct. (%)	Open in both	Time	Fac. open	Cost ($\times 10^3$)	Reduct. (%)	Open in both
0.2	0	53	11	33.3	21.92%	6	459	26	80.96	69.80%	5
0.2	50	13	9	24.28	7.05%	5	109	21	54.75	63.92%	3
0.2	100	27	6	24.51	1.83%	6	113	14	54.57	48.76%	3
0.2	200	103	3	42.85	30.52%	2	737	6	99.43	5.86%	2
0.2	300	260	2	75.76	50.36%	0	8778	4	171.15	23.78%	1
0.1	0	68	12	41.61	30.97%	6	781	32	95.78	71.91%	5
0.1	50	13	10	26.69	21.11%	6	212	24	66.28	74.46%	5
0.1	100	27	6	27.07	3.56%	6	174	16	65.89	63.25%	5
0.1	200	78	3	47.38	26.93%	2	673	8	107.99	12.45%	3
0.1	300	705	2	77.24	49.13%	0	2674	5	188.50	16.07%	2
0.025	0	130	16	60	43.33%	6	1502	42	148.14	72.86%	5
0.025	50	26	13	34.8	40.21%	6	490	34	85.35	75.30%	4
0.025	100	26	10	35.75	16.14%	5	291	22	80.58	73.09%	4
0.025	200	81	4	57.72	15.56%	1	1138	10	129.37	28.60%	4
0.025	300	276	3	109.28	38.52%	2	2977	6	214.73	8.62%	3
0.005	0	134	20	78.33	50.3%	6	1565	50	192.00	77.12%	6
0.005	50	65	15	46.63	51.63%	6	960	42	113.19	83.32%	5
0.005	100	53	11	41.31	33.13%	5	566	26	104.19	79.93%	5
0.005	200	63	5	67.8	6.03%	3	1291	13	158.66	46.53%	5
0.005	300	285	3	108.52	37.88%	2	2025	7	265.32	10.02%	2

Observation 3 explains the impact of the economic point e_j on the solution difficulty. We also note that there is no clear pattern regarding how the computational time varies with respect to ϵ.

Comparing the sets of facilities open in model $\mathcal{P}$ and the UFLP, we find that Observation 4 still holds for Table 6, i.e., between these two sets, the one with less members may not be a subset of the other. In the instance with $|J| = |I| = 100$, $\epsilon = 0.05$, and $e_j = 300$, although both model $\mathcal{P}$ and the UFLP have 7 open facilities, model $\mathcal{P}$ merely chooses 2 among the 7 facilities opened by the UFLP. Furthermore, as shown in Observation 5, Table 6 also demonstrates that model $\mathcal{P}$ achieves very significant savings in the supply chain cost, where the average and maximum percentage cost reduction reach 39.54% and 83.32%, respectively. This verifies the necessity of considering model $\mathcal{P}$ and obtaining its solution for the location and capacity planning problem.

5 Conclusions

In this paper, we devise a branch-and-price approach to study the facility location problem with general nondecreasing facility cost terms. This model provides a unified framework to study more general integrated supply chain design, nonlinear single-sourcing assignment, and many other location problems. By removing the restrictive assumption on the facility cost functions to be concave or submodular in the literature, the general model does not even admit any approximation algorithms unless $P = NP$. The traditional branch-and-price, Lagrangian relaxation, and conic IP reformulation approaches no longer work for this general model either. It is thus of interest to develop an effective approach to address it. To the best of our knowledge, this is the first time that an algorithm is proposed for the facility location problem with general nondecreasing facility cost functions.

We solve the general model using branch-and-price by recasting it as a set-partitioning problem. We show that the pricing problem, which must be solved in each iteration of the column generation procedure for every node on the branch-and-price tree, is NP-hard. By exploiting certain special structures, an optimal solution to the continuous relaxation of the pricing problem is shown to contain at most $\hat{\kappa}$ fractional components, where $\hat{\kappa}$ represents the number of linearly independent attributes of all retailers. In particular, there exists an optimal solution with at most one fractional component if the facility costs are quasi-concave functions. We further study how to identify the $\hat{\kappa}$ variables that may have fractional values and how to determine the binary values for other decision variables. These properties yield a fast implementation of the B&B procedure to obtain an optimal integral solution to the pricing problem. We conduct extensive computational experiments based on the facility location and production/capacity planning models with economies and diseconomies of scale. The computational results establish the effectiveness of the proposed branch-and-price approach, lead to important implications for practice, and demonstrate the importance of studying the general model. The proposed approach thus tackles the computational challenge and facilitates the practical decision making when facing difficult problems that fall into this framework.

Although our model considers the single-sourcing strategy, i.e., each retailer can only be served by one facility, this assumption can be easily relaxed to allow multi-sourcing, where a retailer can be served by multiple facilities. In particular, the proposed algorithm for the relaxation of $\mathcal{Q}$ can be easily applied to obtain a fast solution algorithm for the multi-sourcing counterpart. For instance, Desrochers et al. (1995) develop an approach based on column generation and branch-and-bound for the congested multi-sourcing facility location problem. This column generation and branch-and-bound framework can be applied to the multi-sourcing counterpart of our general model as long as the corresponding pricing problems are solved by our algorithm designed for the relaxation of $\mathcal{Q}$. Note that the existing works considering non-concave facility costs, e.g., Desrochers et al. (1995) and Lu et al. (2014), mainly focus on multi-sourcing. The algorithm in Desrochers et al. (1995) can be applied to the congested single-sourcing location problem if a branch-and-bound scheme is adopted to obtain integral assignments when solving the pricing problems. For the column generation heuristic proposed by Lu et al. (2014) for a location problem with an inverse S-shaped cost, the master problem is derived utilizing the multi-sourcing property and hence could be challenging to be adapted to solve the single-sourcing counterpart.

Acknowledgments

We would like to thank the AE and the referee for their constructive comments that led to this improved version.

References

[1] Ağralı, S., Geunes, J., and Taşkin, Z.C. (2012). A Facility Location Model with Safety Stock Costs: Analysis of the Cost of Single-Sourcing Requirements. *Journal of Global Optimization* **54(3)** 551–581.

[2] Atamtürk, A., Berenguer, G., and Shen, Z.J.M. (2012). A Conic Integer Programming Approach to Stochastic Joint Location-Inventory Problems. *Operations Research* **60(2)** 366–381.

[3] Balinski, M.L. (1965). Integer Programming: Methods, Uses, Computations. *Management Science* **12(3)** 253–313.

[4] Barnhart, C., Johnson, E.L., Nemhauser, G.L., Savelsbergh, M.W.P., and Vance, P.M. (1998). Branch-and-Price: Column Generation for Solving Huge Integer Programs. *Operations Research* **46(3)** 316–329.

[5] Bateni, M. and Hajiaghayi, M. (2012). Assignment Problem in Content Distribution Networks: Unsplittable Hard-Capacitated Facility Location. *ACM Transactions on Algorithms* **8(3)** Article No. 20.

[6] Daskin, M.S. (1995). Network and Discrete Location: Models, Algorithms, and Applications. Wiley-Interscience, New York.

[7] Daskin, M.S., Coullard, C.R., and Shen, Z.J.M. (2002). An Inventory-Location Model: Formulation, Solution Algorithm and Computational Results. *Annals of Operations Research* **110(1-4)** 83–106.

[8] Cornuéjols, G., Nemhauser, G.L., and Wolsey, L.A. (1990). The Uncapacitated Facility Location Problem, In P. Mirchandani and R. Francis, editors, *Discrete Location Theory*, 119–171. John Wiley and Sons Inc.

[9] Desrochers, M., Marcotte, P., and Stan, M. (1995). The Congested Facility Location Problem. *Location Science* **3(1)** 9–23.

[10] Drezner, Z., editor. (1995). *Facility Location: A Survey of Applications and Methods, Springer Series in Operations Research.* New York, NY: Springer Verlag.

[11] Drezner, Z. and Hamacher, H.W., editors. (2002). *Facility Location: Applications and Theory.* Berlin: Springer Verlag.

[12] Eiselt, H.A., and Marianov, V., editors. (2011). *Foundations of Location Analysis, International Series in Operations Research & Management Science.* Boston, MA: Springer.

[13] Federgruen, A. and Groenevelt, H. (1988). M/G/c Queueing Systems with Multiple Customer Classes: Characterization and Control of Achievable Performance Under Nonpreemptive Priority Rules. *Management Science* **34(9)** 1121–1138.

[14] Geunes, J., Shen, M.Z.J., and Romeijn, H.E. (2004). Economic Ordering Decisions with Market Choice Flexibility. *Naval Research Logistics* **51(1)** 117–136.

[15] Hajiaghayi, M., Mahdian, M., and Mirrokni, V.S. (2003). The Facility Location Problem with General Cost Functions. *Networks* **42(1)** 42–47.

[16] Naseraldin, H. and Herer, Y.T. (2008). Integrating the Number and Location of Retail Outlets on a Line with Replenishment Decisions. *Management Science* **54(9)** 1666–1683.

[17] Huang, W., Romeijn, H.E., and Geunes, J. (2005). The Continuous-Time Single-Sourcing Problem with Capacity Expansion Opportunities. *Naval Research Logistics* **52(3)** 193–211.

[18] Kuehn, A.A. and Hamburger, M.J. (1963). A Heuristic Program for Locating Warehouses. *Management Science* **9(4)** 643–666.

[19] Lamiri, M., Xie, X., and Zhang, S. (2008). Column Generation Approach to Operating Theater Planning with Elective and Emergency Patients. *IIE Transactions* **40(9)** 838–852.

[20] Levi, R., Shmoys, D.B., and Swamy, C. (2012). LP-Based Approximation Algorithms for Capacitated Facility Location. *Mathematical Programming* **131(1-2)** 365–379.

[21] Li, Y., Shu, J., Wang, X., Xiu, N., Xu, D., and Zhang, J. (2013). Approximation Algorithms for Integrated Distribution Network Design Problems. *INFORMS Journal on Computing* **25(3)** 572–584.

[22] Lu, D., Gzara, F., and Elhedhli, S. (2014). Facility Location with Economies and Diseconomies of Scale: Models and Column Generation Heuristics. *IIE Transactions* **46(6)** 585–600.

[23] Mak, H.Y. and Shen, Z.J.M. (2009). A Two-Echelon Inventory-Location Problem with Service Considerations. *Naval Research Logistics* **56(8)** 730–744.

[24] Melo, M.T., Nickel, S., and Saldanha-da-Gama, F. (2009). Facility Location and Supply Chain Management - A Review. *European Journal of Operational Research* **196(2)** 401–412.

[25] Ozsen, L., Coullard, C.R., and Daskin, M.S. (2008). Capacitated Warehouse Location Model with Risk Pooling. *Naval Research Logistics* **55(4)** 295–312.

[26] Park, S., Lee, T.-E., and Sung, C.S. (2010). A Three-Level Supply Chain Network Design Model with Risk-Pooling and Lead Times. *Transportation Research Part E* **46(5)** 563–581.

[27] Qi, L., Shen, Z.J.M., and Snyder, L.V. (2010). The Effect of Supply Disruptions on Supply Chain Design Decisions. *Transportation Science* **44(2)** 274–289.

[28] Shen, Z.J.M. (2005). A Multi-Commodity Supply Chain Design Problem. *IIE Transactions* **37(8)** 753–762.

[29] Shen, Z.J.M. (2006). A Profit-Maximizing Supply Chain Network Design Model with Demand Choice Flexibility. *Operations Research Letters* **34(6)** 673–682.

[30] Shen, Z.J.M., Coullard, C.R., and Daskin, M.S. (2003). A Joint Location-Inventory Model. *Transportation Science* **37(1)** 40–55.

[31] Shen, Z.J.M. and Daskin, M.S. (2005). Trade-offs Between Customer Service and Cost in Integrated Supply Chain Design. *Manufacturing & Service Operations Management* **7(3)** 188–207.

[32] Shen, Z.J.M. and Qi, L. (2007). Incorporating Inventory and Routing Costs in Strategic Location Models. *European Journal of Operational Research* **179(2)** 372–389.

[33] Shu, J. (2010). An Efficient Greedy Heuristic for Warehouse-Retailer Network Design Optimization. *Transportation Science* **44(2)** 183–192.

[34] Shu, J., Teo, C.-P., and Shen, Z.J.M. (2005). Stochastic Transportation-Inventory Network Design Problem. *Operations Research* **53(1)** 48–60.

[35] Snyder, L.V., Daskin, M.S., and Teo, C.-P. (2007). The Stochastic Location Model with Risk Pooling. *European Journal of Operational Research* **179(3)** 1221–1238.

[36] Sourirajan, K., Ozsen, L., and Uzsoy, R. (2007). A Single-Product Network Design Model with Lead Time and Safety Stock Considerations. *IIE Transactions* **39(5)** 411–424.

[37] Strinka, Z.M.A., Romeijn, H.R., and Wu, J. (2013). Exact and Heuristic Methods for a Class of Selective Newsvendor Problems with Normally Distributed Demands. *OMEGA* **41(2)** 250–258.

[38] Teo, C.-P. and Shu, J. (2004). Warehouse-Retailer Network Design Problem. *Operations Research* **52(3)** 396–408.

[39] Vidyarthi, N., Çelebi, E., Elhedhli, S., and Jewkes, E. (2007). Integrated Production-Inventory-Distribution System Design with Risk Pooling: Model Formulation and Heuristic Solution. *Transportation Science* **41(3)** 392–408.

Online Supplement

A Branch-and-Price Algorithm for Facility Location with General Facility Cost Functions

Wenjun Ni* Jia Shu† Miao Song‡ Dachuan Xu§ and Kaike Zhang¶

March 26, 2021

Appendix A. Proofs

This section provides the proofs of all the theorems and propositions in this paper.

A.1 Proof of Theorem 1

Consider the knapsack problem

$$\max\left\{\sum_{i=1}^{n} a_i z_i \;\middle|\; \sum_{i=1}^{n} b_i z_i \leq W,\ z_i \in \{0,1\},\ \forall i \in I\right\}$$

where a_i, b_i, and W are positive integers. Define an instance of $\mathcal{Q}$ with $\kappa = 1$ and

$$\Phi(x) = \begin{cases} 0, & \text{if } x \in [0, W] \\ \left(\sum_{i=1}^{n} a_i + 1\right)(x - W), & \text{if } x > W \end{cases}$$

*Department of Management Science and Engineering, School of Economics and Management, Southeast University. Email: wjni@seu.edu.cn

†School of Economics and Management, Southwest Jiaotong University, Chengdu, Sichuan 610031, PR China. Email:jshu@seu.edu.cn

‡Department of Logistics and Maritime Studies, Faculty of Business, The Hong Kong Polytechnic University. Email: miao.song@polyu.edu.hk

§Department of Applied Mathematics, Beijing University of Technology. Email: xudc@bjut.edu.cn.

¶Department of Management Science and Engineering, School of Economics and Management, Southeast University. Email: kaikezhang@utk.edu

for any $x \geq 0$. Consider a feasible solution $\mathbf{z}$ to $\mathcal{Q}$ such that $\sum_{i=1}^n b_i z_i > W$. Because b_i, z_i, and W are integers, we have $\sum_{i=1}^n b_i z_i - W \geq 1$ and hence

$$-\sum_{i=1}^n a_i z_i + \Phi\left(\sum_{i=1}^n b_i z_i\right) = -\sum_{i=1}^n a_i z_i + \left(\sum_{i=1}^n a_i + 1\right)\left(\sum_{i=1}^n b_i z_i - W\right)$$
$$\geq -\sum_{i=1}^n a_i z_i + \left(\sum_{i=1}^n a_i + 1\right) \geq 1.$$

Note that the solution $\bar{\mathbf{z}}$ where $\bar{z}_i = 0$ for any $i \in I$ is also feasible to $\mathcal{Q}$ and its corresponding objective value is zero. Therefore, any solution $\mathbf{z}$ such that $\sum_{i=1}^n b_i z_i > W$ is not optimal to $\mathcal{Q}$, and hence this instance of $\mathcal{Q}$ is equivalent to

$$\min\left\{-\sum_{i=1}^n a_i z_i + \Phi\left(\sum_{i=1}^n b_i z_i\right) \,\middle|\, \sum_{i=1}^n b_i z_i \leq W,\ z_i \in \{0,1\}\ \forall i \in I\right\}$$

Because $\Phi(\sum_{i=1}^n b_i z_i) = 0$ if $\sum_{i=1}^n b_i z_i \leq W$, it is straightforward that this problem is equivalent to the knapsack problem, which proves the NP-hardness of $\mathcal{Q}$. $\square$

A.2 Proof of Proposition 1

Consider $i^*, i' \in I$ such that $a_{i'} < a_{i^*}$ and $\mathbf{b}_{i'} \geq \mathbf{b}_{i^*}$. Assume for contradiction that there exists an optimal solution $\mathbf{z}^*$ to $\mathcal{Q}$ such that $z_{i^*}^* = 0$ and $z_{i'}^* = 1$. Define $\bar{\mathbf{z}}$ such that $\bar{z}_{i^*} = 1$, $\bar{z}_{i'} = 0$, and $\bar{z}_i = z_i^*$ for all $i \in I \setminus \{i^*, i'\}$. Then

$$-\sum_{i=1}^n a_i \bar{z}_i + \Phi\left(\sum_{i=1}^n \mathbf{b}_i \bar{z}_i\right) = -\sum_{i=1}^n a_i z_i^* + a_{i'} - a_{i^*} + \Phi\left(\sum_{i=1}^n \mathbf{b}_i z_i^* - \mathbf{b}_{i'} + \mathbf{b}_{i^*}\right)$$
$$< -\sum_{i=1}^n a_i z_i^* + \Phi\left(\sum_{i=1}^n \mathbf{b}_i z_i^*\right)$$

where the equality follows from the definition of $\bar{\mathbf{z}}$ and the inequality follows from $a_{i'} < a_{i^*}$, $b_{i'} \geq \mathbf{b}_{i^*}$, and the property that $\Phi(\cdot)$ is nondecreasing. This contradicts the optimality of $\mathbf{z}^*$ and hence part (i) must be true. Part (ii) can be obtained by a similar argument. $\square$

A.3 Proof of Theorem 2

Suppose that $(z_i^*\ \forall i \in I)$ is an optimal solution to the relaxation of $\mathcal{Q}$. Consider the following linear programming problem:

$$\mathcal{P}_I: \quad \max\left\{\sum_{i\in I} a_i z_i \,\middle|\, \sum_{i\in I} \hat{\mathbf{b}}_i z_i = \sum_{i\in I} \hat{\mathbf{b}}_i z_i^*,\ z_i \in [0,1]\ \forall i \in I\right\}.$$

Obviously, $(z_i^* \,\forall i \in I)$ is feasible to $\mathcal{P}_I$. As the feasible region of $\mathcal{P}_I$ is bounded, $\mathcal{P}_I$ has an optimal extreme point $(\bar{z}_i \,\forall i \in I)$. Note that $\mathcal{P}_I$ has $\hat{\kappa}$ equality constraints, which means that at least $|I| - \hat{\kappa}$ of the $z_i \in [0,1]$ constraints must be tight at $(\bar{z}_i \,\forall i \in I)$. Therefore, the set $\{i \in I : \bar{z}_i \in \{0,1\}\}$ has at least $|I| - \hat{\kappa}$ members, i.e., $(\bar{z}_i \,\forall i \in I)$ has at most $\hat{\kappa}$ fractional components.

Also note that $(\bar{z}_i \,\forall i \in I)$ is feasible to the relaxation of $\mathcal{Q}$. Since $(\bar{z}_i \,\forall i \in I)$ is optimal to $\mathcal{P}_I$ whereas $(z_i^* \,\forall i \in I)$ is feasible to $\mathcal{P}_I$, we have $\sum_{i\in I} a_i \bar{z}_i \geq \sum_{i \in I} a_i z_i^*$. The feasibility of $(\bar{z}_i \,\forall i \in I)$ to $\mathcal{P}_I$ implies that $\sum_{i\in I} b_i^k \bar{z}_i = \sum_{i\in I} b_i^k z_i^*$ for any $k \in \{1, ..., \hat{\kappa}\}$. For any $k \in \{\hat{\kappa}+1, ..., \kappa\}$, (5) yields

$$\sum_{i\in I} b_i^k \bar{z}_i = \sum_{i\in I} \left(\mathbf{p}_k^T \hat{\mathbf{b}}_i\right) \bar{z}_i = \mathbf{p}_k^T \left(\sum_{i\in I} \hat{\mathbf{b}}_i \bar{z}_i\right) = \mathbf{p}_k^T \left(\sum_{i\in I} \hat{\mathbf{b}}_i z_i^*\right) = \sum_{i\in I} \left(\mathbf{p}_k^T \hat{\mathbf{b}}_i\right) z_i^* = \sum_{i\in I} b_i^k z_i^*.$$

Therefore, $(\bar{z}_i \,\forall i \in I)$ is also optimal to the relaxation of $\mathcal{Q}$. □

A.4 Proof of Proposition 2

WLOG, suppose that $(z_i^* \,\forall i \in I)$ is an optimal solution to the relaxation of $\mathcal{Q}$, where $z_1^*, z_2^* \in (0,1)$. It suffices to show that the relaxation of $\mathcal{Q}$ has an optimal solution $(\bar{z}_i \,\forall i \in I)$ such that (i) $\bar{z}_i = z_i^*$ for all $i \in I \setminus \{1,2\}$ and (ii) either $\bar{z}_1$ or $\bar{z}_2$ is integral.

Define

$$\Phi_{1,2}(z_1, z_2) \equiv \Phi\left(\mathbf{b}_1 z_1 + \mathbf{b}_2 z_2 + \sum_{i\in I\setminus\{1,2\}} \mathbf{b}_i z_i^*\right).$$

For any $(\dot{z}_1, \dot{z}_2), (\ddot{z}_1, \ddot{z}_2) \in \mathbb{R}^2$ and $\lambda \in [0,1]$,

$$\begin{aligned}
&\Phi_{1,2}\Big(\lambda \dot{z}_1 + (1-\lambda)\ddot{z}_1, \lambda \dot{z}_2 + (1-\lambda)\ddot{z}_2\Big) = \Phi\left(\sum_{i=1}^{2} \mathbf{b}_i\Big(\lambda \dot{z}_i + (1-\lambda)\ddot{z}_i\Big) + \sum_{i\in I\setminus\{1,2\}} \mathbf{b}_i z_i^*\right) \\
&= \Phi\left(\lambda\left(\sum_{i=1}^{2} \mathbf{b}_i \dot{z}_i + \sum_{i\in I\setminus\{1,2\}} \mathbf{b}_i z_i^*\right) + (1-\lambda)\left(\sum_{i=1}^{2} \mathbf{b}_i \ddot{z}_i + \sum_{i\in I\setminus\{1,2\}} \mathbf{b}_i z_i^*\right)\right) \\
&\geq \min\left\{\Phi\left(\sum_{i=1}^{2} \mathbf{b}_i \dot{z}_i + \sum_{i\in I\setminus\{1,2\}} \mathbf{b}_i z_i^*\right), \Phi\left(\sum_{i=1}^{2} \mathbf{b}_i \ddot{z}_i + \sum_{i\in I\setminus\{1,2\}} \mathbf{b}_i z_i^*\right)\right\} \\
&= \min\Big\{\Phi_{1,2}(\dot{z}_1, \dot{z}_2), \Phi_{1,2}(\ddot{z}_1, \ddot{z}_2)\Big\},
\end{aligned}$$

where the inequality follows from the quasi-concavity of $\Phi(\cdot)$. Therefore, the function $\Phi_{1,2}(\cdot)$ is also quasi-concave. Furthermore, as $\mathbf{b}_i \geq \mathbf{0}$ for $i \in \{1,2\}$, the monotonicity of $\Phi(\cdot)$ yields that $\Phi_{1,2}(\cdot)$ is increasing in z_1 and z_2 in $[0,1]$, respectively.

Consider the following problem:

$$\mathcal{P}_{1,2}: \quad v_{1,2}^* = \max_{z_1, z_2} \Big\{ a_1 z_1 + a_2 z_2 : \Phi_{1,2}(z_1, z_2) \le \Phi_{1,2}(z_1^*, z_2^*),\ z_1 \in [0,1],\ z_2 \in [0,1] \Big\}.$$

Let $(\bar{z}_1, \bar{z}_2)$ be an optimal solution to $\mathcal{P}_{1,2}$. Set $\bar{z}_i = z_i^*$ for all $i \in I \setminus \{1,2\}$. As (z_1^*, z_2^*) is feasible to $\mathcal{P}_{1,2}$ and $(\bar{z}_1, \bar{z}_2)$ is optimal to $\mathcal{P}_{1,2}$, we have

$$-\sum_{i=1}^{n} a_i \bar{z}_i + \Phi\left(\sum_{i=1}^{n} \mathbf{b}_i \bar{z}_i\right) = -\left(a_1 \bar{z}_1 + a_2 \bar{z}_2 + \sum_{i \in I \setminus \{1,2\}} a_i z_i^*\right) + \Phi_{1,2}(\bar{z}_1, \bar{z}_2)$$
$$\le -\left(a_1 z_1^* + a_2 z_2^* + \sum_{i \in I \setminus \{1,2\}} a_i z_i^*\right) + \Phi_{1,2}(z_1^*, z_2^*) = -\sum_{i=1}^{n} a_i z_i^* + \Phi\left(\sum_{i=1}^{n} \mathbf{b}_i z_i^*\right).$$

Obviously, $(\bar{z}_i\ \forall i \in I)$ is feasible to the relaxation of $\mathcal{Q}$. Thus, $(\bar{z}_i\ \forall i \in I)$ is also optimal to the relaxation of $\mathcal{Q}$. As a result, we can complete the proof by showing that $\mathcal{P}_{1,2}$ has an optimal solution with at most one fractional component.

WLOG, we assume $\Phi_{1,2}(1,0) \le \Phi_{1,2}(0,1)$. The monotonicity of $\Phi_{1,2}(\cdot)$ implies $\Phi_{1,2}(0,0) \le \Phi_{1,2}(1,0) \le \Phi_{1,2}(0,1) \le \Phi_{1,2}(1,1)$. Also note that $z_1^*, z_2^* \in (0,1)$, which yields $\Phi_{1,2}(0,0) \le \Phi_{1,2}(z_1^*, z_2^*) \le \Phi_{1,2}(1,1)$. It is sufficient to consider the following four cases:

Case 1. Suppose $\Phi_{1,2}(z_1^*, z_2^*) = \Phi_{1,2}(1,1)$. Then $\Phi_{1,2}(z_1, z_2) \le \Phi_{1,2}(z_1^*, z_2^*)$ for all $z_1, z_2 \in [0,1]$. The problem $\mathcal{P}_{1,2}$ is reduced to $\max\{a_1 z_1 + a_2 z_2 : z_1 \in [0,1],\ z_2 \in [0,1]\}$. Therefore, the set $\{(0,0), (1,0), (0,1), (1,1)\}$ must contain an optimal solution of $\mathcal{P}_{1,2}$.

Case 2. Suppose $\Phi_{1,2}(0,1) \le \Phi_{1,2}(z_1^*, z_2^*) < \Phi_{1,2}(1,1)$. According to the monotonicity of $\Phi_{1,2}(\cdot)$, we can define

$$\begin{aligned} \bar{z}_1 &= \max\{z_1 : \Phi_{1,2}(z_1, 1) \le \Phi_{1,2}(z_1^*, z_2^*)\} \in [0,1) \\ \bar{z}_2 &= \max\{z_2 : \Phi_{1,2}(1, z_2) \le \Phi_{1,2}(z_1^*, z_2^*)\} \in [0,1). \end{aligned} \tag{10}$$

Let $\mathcal{Z}$ and $CH(\mathcal{Z})$ denote the set $\{(0,0), (1,0), (1, \bar{z}_2), (\bar{z}_1, 1), (0,1)\}$ and the convex hull constructed by the points in $\mathcal{Z}$, respectively (cf. Case 2 of Figure 2). Consider any $(z_1, z_2) \in [0,1]^2 \setminus CH(\mathcal{Z})$. As shown in Case 2 of Figure 2, it can be represented by a convex combination of $\{(z_1', 1), (1,1), (1, z_2')\}$, where $z_1' \in (\bar{z}_1, 1]$ and $z_2' \in (\bar{z}_2, 1]$. Applying the quasi-concavity of $\Phi_{1,2}(\cdot)$, we have

$$\Phi_{1,2}(z_1, z_2) \ge \min\Big\{\Phi_{1,2}(z_1', 1), \Phi_{1,2}(1, z_2'), \Phi_{1,2}(1,1)\Big\}.$$

According to the definition of $\bar{z}_1$ and $\bar{z}_2$, as $z_1' \in (\bar{z}_1, 1]$ and $z_2' \in (\bar{z}_2, 1]$, we obtain $\Phi_{1,2}(z_1', 1) > \Phi_{1,2}(z_1^*, z_2^*)$ and $\Phi_{1,2}(1, z_2') > \Phi_{1,2}(z_1^*, z_2^*)$ because $\Phi_{1,2}(\cdot)$ is an increasing function. Combining with $\Phi_{1,2}(z_1^*, z_2^*) < \Phi_{1,2}(1,1)$, we have

$$\Phi_{1,2}(z_1, z_2) > \Phi_{1,2}(z_1^*, z_2^*) \quad \forall (z_1, z_2) \in [0,1]^2 \setminus CH(\mathcal{Z}),$$

i.e.,

$$\Big\{(z_1, z_2) : \Phi_{1,2}(z_1, z_2) \le \Phi_{1,2}(z_1^*, z_2^*),\ z_1 \in [0,1],\ z_2 \in [0,1]\Big\} \subseteq CH(\mathcal{Z}).$$

Hence,

$$v_{1,2}^* \le \max_{z_1,z_2}\Big\{a_1 z_1 + a_2 z_2 : (z_1, z_2) \in CH(\mathcal{Z})\Big\} = \max_{z_1,z_2}\Big\{a_1 z_1 + a_2 z_2 : (z_1, z_2) \in \mathcal{Z}\Big\},$$

where the equality is obtained as $a_1 z_1 + a_2 z_2$ is linear in (z_1, z_2) and $CH(\mathcal{Z})$ is the convex hull of $\mathcal{Z}$. Note that all the points in $\mathcal{Z}$ are feasible to $\mathcal{P}_{1,2}$. Therefore, an optimal solution to $\mathcal{P}_{1,2}$ is contained in $\mathcal{Z}$, whose members have at most one fractional component.

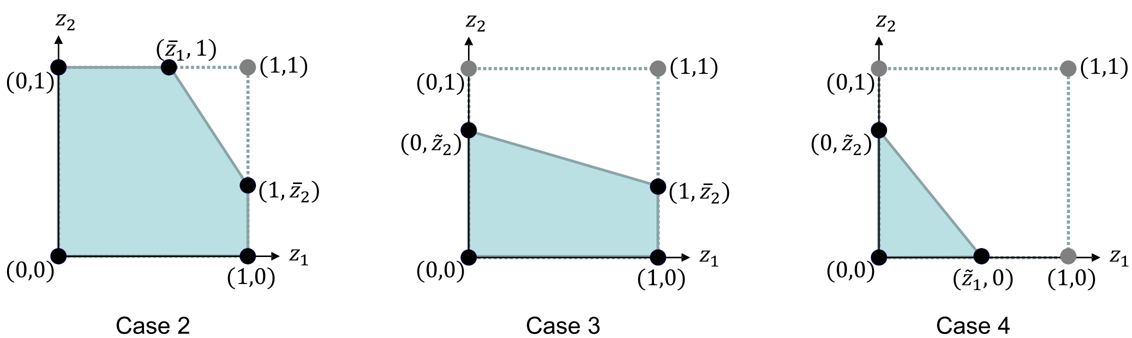

Figure 2: Optimal Solution to the Problem $\mathcal{P}_{1,2}$

Case 3. Suppose $\Phi_{1,2}(1, 0) \le \Phi_{1,2}(z_1^*, z_2^*) < \Phi_{1,2}(0, 1)$. Consider $\bar{z}_2$ defined in (10) and define

$$\tilde{z}_2 = \max\{z_2 : \Phi_{1,2}(0, z_2) \le \Phi_{1,2}(z_1^*, z_2^*)\} \in [0, 1). \tag{11}$$

Let $\mathcal{Z}$ and $CH(\mathcal{Z})$ denote the set $\{(0, 0), (1, 0), (1, \bar{z}_2), (0, \tilde{z}_2)\}$ and the convex hull constructed by the points in $\mathcal{Z}$, respectively (cf. Case 3 of Figure 2). Any $(z_1, z_2) \in [0, 1]^2 \setminus CH(\mathcal{Z})$ can be represented by a convex combination of $\{(1, z_2'), (1, 1), (0, 1), (0, z_2'')\}$, where $z_2' \in (\bar{z}_2, 1]$ and $z_2'' \in (\tilde{z}_2, 1]$. Combining with the definitions of $\bar{z}_2$ and $\tilde{z}_2$, the quasi-concavity and monotonicity of $\Phi_{1,2}(\cdot)$ yield

$$\Phi_{1,2}(z_1, z_2) \ge \min\Big\{\Phi_{1,2}(1, z_2'), \Phi_{1,2}(1, 1), \Phi_{1,2}(0, 1), \Phi_{1,2}(0, z_2'')\Big\} > \Phi_{1,2}(z_1^*, z_2^*).$$

By the same argument in Case 2, we can establish that $\mathcal{P}_{1,2}$ has an optimal solution in $\mathcal{Z}$ with at most one fractional component.

Case 4. Suppose $\Phi_{1,2}(0, 0) \le \Phi_{1,2}(z_1^*, z_2^*) < \Phi_{1,2}(1, 0)$. Consider $\tilde{z}_2$ defined in (11) and define

$$\tilde{z}_1 = \max\{z_1 : \Phi_{1,2}(z_1, 0) \le \Phi_{1,2}(z_1^*, z_2^*)\} \in [0, 1).$$

Let $\mathcal{Z}$ be the set $\{(0, 0), (0, \tilde{z}_2), (\tilde{z}_1, 0)\}$ (cf. Case 4 of Figure 2). Similar to Cases 2 and 3, we can show that $\mathcal{Z}$ contains an optimal solution to $\mathcal{P}_{1,2}$ with at most one fractional component. □

A.5 Proof of Proposition 3

It is equivalent to show that $P \neq \emptyset$ where

$$P = \left\{ \boldsymbol{\alpha} \geq \mathbf{0} \;\middle|\; \begin{array}{ll} \boldsymbol{\alpha}^T \mathbf{b}_i / a_i \leq 1 & \forall\, i \text{ such that } z_i^* > 0 \\ \boldsymbol{\alpha}^T \mathbf{b}_i / a_i \geq 1 & \forall\, i \text{ such that } z_i^* < 1 \end{array} \right\}.$$

According to Farkas' Lemma, exactly one of the following two statements is true: (i) $P \neq \emptyset$ or (ii) $D \neq \emptyset$ where

$$D = \left\{ \begin{array}{ll} \hat{\lambda}_i \geq 0 & \forall\, i \text{ such that } z_i^* > 0 \\ \hat{\mu}_i \geq 0 & \forall\, i \text{ such that } z_i^* < 1 \end{array} \;\middle|\; \begin{array}{l} \sum_{i:z_i^*>0} \hat{\lambda}_i - \sum_{i:z_i^*<1} \hat{\mu}_i < 0 \\ \sum_{i:z_i^*>0} \frac{\mathbf{b}_i}{a_i} \hat{\lambda}_i - \sum_{i:z_i^*<1} \frac{\mathbf{b}_i}{a_i} \hat{\mu}_i \geq \mathbf{0} \end{array} \right\}.$$

Therefore, it suffices to prove $D = \emptyset$. Let us assume for contradiction that there exists some $(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{\mu}}) \in D$. Consider $(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}})$ such that $\tilde{\lambda}_i = \hat{\lambda}_i$ for any i such that $z_i^* = 1$, $\tilde{\mu}_i = \hat{\mu}_i$ for any i such that $z_i^* = 0$, and

$$\tilde{\lambda}_i = \max\{\hat{\lambda}_i - \hat{\mu}_i, 0\} \qquad \text{and} \qquad \tilde{\mu}_i = \max\{\hat{\mu}_i - \hat{\lambda}_i, 0\} \tag{12}$$

for any i such that $z_i^* \in (0, 1)$. Obviously, $\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}} \geq \mathbf{0}$. Also note that

$$\begin{aligned}
\sum_{i:z_i^*>0} \hat{\lambda}_i - \sum_{i:z_i^*<1} \hat{\mu}_i &= \sum_{i:z_i^*=1} \hat{\lambda}_i + \sum_{i:z_i^*\in(0,1)} (\hat{\lambda}_i - \hat{\mu}_i) - \sum_{i:z_i^*=0} \hat{\mu}_i \\
&= \sum_{i:z_i^*=1} \tilde{\lambda}_i + \sum_{i:z_i^*\in(0,1)} (\tilde{\lambda}_i - \tilde{\mu}_i) - \sum_{i:z_i^*=0} \tilde{\mu}_i = \sum_{i:z_i^*>0} \tilde{\lambda}_i - \sum_{i:z_i^*<1} \tilde{\mu}_i, \\
\sum_{i:z_i^*>0} \frac{\mathbf{b}_i}{a_i} \hat{\lambda}_i - \sum_{i:z_i^*<1} \frac{\mathbf{b}_i}{a_i} \hat{\mu}_i &= \sum_{i:z_i^*=1} \frac{\mathbf{b}_i}{a_i} \hat{\lambda}_i + \sum_{i:z_i^*\in(0,1)} \frac{\mathbf{b}_i}{a_i} (\hat{\lambda}_i - \hat{\mu}_i) - \sum_{i:z_i^*=0} \frac{\mathbf{b}_i}{a_i} \hat{\mu}_i \\
&= \sum_{i:z_i^*=1} \frac{\mathbf{b}_i}{a_i} \tilde{\lambda}_i + \sum_{i:z_i^*\in(0,1)} \frac{\mathbf{b}_i}{a_i} (\tilde{\lambda}_i - \tilde{\mu}_i) - \sum_{i:z_i^*=0} \frac{\mathbf{b}_i}{a_i} \tilde{\mu}_i \\
&= \sum_{i:z_i^*>0} \frac{\mathbf{b}_i}{a_i} \tilde{\lambda}_i - \sum_{i:z_i^*<1} \frac{\mathbf{b}_i}{a_i} \tilde{\mu}_i,
\end{aligned}$$

which indicates that $(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}}) \in D$. Consider the following two cases:

Case 1. Suppose that $\sum_{i:z_i^*>0} \tilde{\lambda}_i = 0$, which implies that $\tilde{\lambda}_i = 0$ for any i such that $z_i^* > 0$. The definition of D yields that $\sum_{i:z_i^*<1} \frac{\mathbf{b}_i}{a_i} \tilde{\mu}_i = \mathbf{0}$, i.e., $\sum_{i:z_i^*<1} \frac{b_i^k}{a_i} \tilde{\mu}_i = 0$ for all $k = 1, ..., \kappa$, implying

$$\sum_{i:z_i^*<1} \frac{\sum_{k=1}^{\kappa} b_i^k}{a_i} \tilde{\mu}_i = \sum_{i:z_i^*<1} \sum_{k=1}^{\kappa} \frac{b_i^k}{a_i} \tilde{\mu}_i = \sum_{k=1}^{\kappa} \sum_{i:z_i^*<1} \frac{b_i^k}{a_i} \tilde{\mu}_i = 0.$$

Recall that $a_i > 0$, $\mathbf{b}_i \geq \mathbf{0}$, and $\max\{b_i^k : k = 1, ..., \kappa\} > 0$ for all i. We have $\tilde{\mu}_i = 0$ for any i such that $z_i^* < 1$ and so $\sum_{i:z_i^*<1} \tilde{\mu}_i = 0$, which contradicts the first constraint in the definition of D.

Case 2. Suppose that $\sum_{i:z_i^*>0}\tilde{\lambda}_i > 0$, which, by the first constraint in the definition of D, yields $\sum_{i:z_i^*<1}\tilde{\mu}_i > 0$. Let

$$\lambda_i = \frac{\tilde{\lambda}_i}{\sum_{i':z_{i'}^*>0}\tilde{\lambda}_{i'}} \ \forall\ i \text{ such that } z_i^* > 0 \qquad \text{and} \qquad \mu_i = \frac{\tilde{\mu}_i}{\sum_{i':z_{i'}^*<1}\tilde{\mu}_{i'}} \ \forall\ i \text{ such that } z_i^* < 1.$$

(12) yields that either $\lambda_i = 0$ or $\mu_i = 0$ for any i such that $z_i^* \in (0,1)$. Therefore, we can define $S_1 = \{i : \lambda_i > 0, z_i^* > 0\}$ and $S_0 = \{i : \mu_i > 0, z_i^* < 1\}$ where $S_1 \cap S_0 = \emptyset$. Obviously,

$$\sum_{i\in S_1}\lambda_i = \sum_{i:z_i^*>0}\lambda_i = \sum_{i:z_i^*>0}\frac{\tilde{\lambda}_i}{\sum_{i':z_{i'}^*>0}\tilde{\lambda}_{i'}} = 1 \quad \text{and} \quad \sum_{i\in S_0}\mu_i = \sum_{i:z_i^*<1}\mu_i = \sum_{i:z_i^*<1}\frac{\tilde{\mu}_i}{\sum_{i':z_{i'}^*<1}\tilde{\mu}_{i'}} = 1, \tag{13}$$

which also suggests $S_0, S_1 \neq \emptyset$.

The feasibility of $(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}})$ yields

$$\sum_{i:z_i^*>0}\frac{\mathbf{b}_i}{a_i}\tilde{\lambda}_i - \sum_{i:z_i^*<1}\frac{\mathbf{b}_i}{a_i}\tilde{\mu}_i \geq \mathbf{0}.$$

Applying the definitions of $\boldsymbol{\lambda}$ and $\boldsymbol{\mu}$, we obtain

$$\sum_{i:z_i^*>0}\frac{\mathbf{b}_i}{a_i}\left(\sum_{i':z_{i'}^*>0}\tilde{\lambda}_{i'}\lambda_i\right) - \sum_{i:z_i^*<1}\frac{\mathbf{b}_i}{a_i}\left(\sum_{i':z_{i'}^*<1}\tilde{\mu}_{i'}\mu_i\right) \geq \mathbf{0} \quad \Rightarrow \quad \sum_{i:z_i^*>0}\frac{\mathbf{b}_i}{a_i}\lambda_i \geq \frac{\sum_{i':z_{i'}^*<1}\tilde{\mu}_{i'}}{\sum_{i':z_{i'}^*>0}\tilde{\lambda}_{i'}}\sum_{i:z_i^*<1}\frac{\mathbf{b}_i}{a_i}\mu_i.$$

As $(\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}}) \in D$, the first constraint in the definition of D implies that

$$\sum_{i:z_i^*>0}\tilde{\lambda}_i - \sum_{i:z_i^*<1}\tilde{\mu}_i < 0 \quad \Rightarrow \quad \frac{\sum_{i:z_i^*<1}\tilde{\mu}_i}{\sum_{i:z_i^*>0}\tilde{\lambda}_i} > 1.$$

Note that $\tilde{\boldsymbol{\lambda}}, \tilde{\boldsymbol{\mu}} \geq \mathbf{0}$ implies $\boldsymbol{\lambda}, \boldsymbol{\mu} \geq \mathbf{0}$. Also recall that $a_i > 0$ and $\mathbf{b}_i \geq \mathbf{0}$ for all i. Combining with the definitions of S_1 and S_0, we obtain

$$\sum_{i\in S_1}\frac{\mathbf{b}_i}{a_i}\lambda_i = \sum_{i:z_i^*>0}\frac{\mathbf{b}_i}{a_i}\lambda_i \geq \sum_{i:z_i^*<1}\frac{\mathbf{b}_i}{a_i}\mu_i = \sum_{i\in S_0}\frac{\mathbf{b}_i}{a_i}\mu_i \geq \mathbf{0}$$

and

$$\sum_{i\in S_1}\frac{b_i^k}{a_i}\lambda_i > \sum_{i\in S_0}\frac{b_i^k}{a_i}\mu_i \qquad \forall\ k \in \{1, ..., \kappa\} \text{ such that } \sum_{i\in S_0}\frac{b_i^k}{a_i}\mu_i > 0.$$

To simplify the notation, let

$$\mathbf{x}_1 \equiv \sum_{i\in S_1}\frac{\mathbf{b}_i}{a_i}\lambda_i \qquad \text{and} \qquad \mathbf{x}_0 \equiv \sum_{i\in S_0}\frac{\mathbf{b}_i}{a_i}\mu_i. \tag{14}$$

As $a_i > 0$ and $\mathbf{b}_i \neq \mathbf{0}$ for all i, $\mu_i > 0$ for all $i \in S_0$, and $S_0 \neq \emptyset$, there exists at least one $k \in \{1, ..., \kappa\}$ such that $x_0^k \neq 0$. The above analysis then yields that $\mathbf{x}_1 \geq \mathbf{x}_0 \geq \mathbf{0}$ and $x_1^k > x_0^k > 0$ for some $k \in \{1, ..., \kappa\}$. Let

$$\epsilon \equiv \max \left\{ \frac{x_0^k}{x_1^k} : x_1^k > 0, k = 1, ..., \kappa \right\} \in [0, 1).$$

Furthermore, define

$$\delta \equiv \min \left\{ \min \left\{ \frac{a_i}{\lambda_i} z_i^* : i \in S_1 \right\}, \min \left\{ \frac{a_i}{\mu_i}(1 - z_i^*) : i \in S_0 \right\} \right\}.$$

The definitions of S_1 and S_0 imply that $\delta \in (0, +\infty)$.

Consider the solution $(\bar{z}_i \, \forall i \in I)$ such that $\bar{z}_i = z_i^* - \frac{\lambda_i}{a_i} \epsilon \delta$ for any $i \in S_1$, $\bar{z}_i = z_i^* + \frac{\mu_i}{a_i} \delta$ for any $i \in S_0$, and $\bar{z}_i = z_i^*$ for any $i \notin S_1 \cup S_0$. Applying the definitions of ϵ and δ, it is straightforward to verify that $(\bar{z}_i \, \forall i \in I)$ is feasible to the relaxation of $\mathcal{Q}$. Note that

$$\begin{aligned}
\sum_{i \in I} a_i \bar{z}_i &= \sum_{i \in S_1} a_i \left(z_i^* - \frac{\lambda_i}{a_i} \epsilon \delta \right) + \sum_{i \in S_0} a_i \left(z_i^* + \frac{\mu_i}{a_i} \delta \right) + \sum_{i \notin S_1 \cup S_0} a_i z_i^* \\
&= \sum_{i \in I} a_i z_i^* - \sum_{i \in S_1} \lambda_i \epsilon \delta + \sum_{i \in S_0} \mu_i \delta = \sum_{i \in I} a_i z_i^* - \epsilon \delta + \delta > \sum_{i \in I} a_i z_i^*,
\end{aligned}$$

where the third equality follows from (13) and the inequality follows from $\delta > 0$ and $0 \leq \epsilon < 1$. For any $k \in \{1, ..., \kappa\}$,

$$\begin{aligned}
\sum_{i \in I} b_i^k \bar{z}_i &= \sum_{i \in S_1} b_i^k \left(z_i^* - \frac{\lambda_i}{a_i} \epsilon \delta \right) + \sum_{i \in S_0} b_i^k \left(z_i^* + \frac{\mu_i}{a_i} \delta \right) + \sum_{i \notin S_1 \cup S_0} b_i^k z_i^* \\
&= \sum_{i \in I} b_i^k z_i^* - \sum_{i \in S_1} \frac{b_i^k}{a_i} \lambda_i \epsilon \delta + \sum_{i \in S_0} \frac{b_i^k}{a_i} \mu_i \delta = \sum_{i \in I} b_i^k z_i^* - x_1^k \epsilon \delta + x_0^k \delta,
\end{aligned}$$

where the third equality follows from (14). If $x_1^k = 0$, then $x_0^k = 0$ as $x_0^k \leq x_1^k$ and hence $-x_1^k \epsilon \delta + x_0^k \delta = 0$; otherwise, the definition of ϵ yields

$$-x_1^k \epsilon \delta + x_0^k \delta \leq -x_1^k \frac{x_0^k}{x_1^k} \delta + x_0^k \delta = 0.$$

Thus, we obtain $\sum_{i \in I} b_i^k \bar{z}_i \leq \sum_{i \in I} b_i^k z_i^*$ for all $k \in \{1, ..., \kappa\}$. Applying the monotonicity of $\Phi(\cdot)$, we can show that $(\bar{z}_i \, \forall i \in I)$ has a smaller objective value than that of $(z_i^* \, \forall i \in I)$, which contradicts the optimality of $(z_i^* \, \forall i \in I)$. □

A.6 Proof of Proposition 4

According to Proposition 3, it is sufficient to show that for any $\boldsymbol{\alpha} \geq \mathbf{0}$, there exists some $\boldsymbol{\beta}$ considered in Line 11 of Algorithm 1 such that $S_1(\boldsymbol{\beta}) \subseteq S_1(\boldsymbol{\alpha})$, $S_0(\boldsymbol{\beta}) \subseteq S_0(\boldsymbol{\alpha})$, and $H(\boldsymbol{\beta}) \supseteq H(\boldsymbol{\alpha})$, where $S_1(\cdot)$, $S_0(\cdot)$, and $H(\cdot)$ are defined in (6).

For any $\boldsymbol{\gamma} = [\gamma^1, \gamma^2, ..., \gamma^\kappa]^T$, let $\hat{\boldsymbol{\gamma}} = [\gamma^1, \gamma^2, ..., \gamma^{\hat{\kappa}}]^T$. Applying (5), we have

$$\boldsymbol{\gamma}^T \mathbf{b}_i = \hat{\boldsymbol{\gamma}}^T \hat{\mathbf{b}}_i + \sum_{k=\hat{\kappa}+1}^{\kappa} \gamma^k b_i^k = \hat{\boldsymbol{\gamma}}^T \hat{\mathbf{b}}_i + \sum_{k=\hat{\kappa}+1}^{\kappa} \gamma^k \mathbf{p}_k^T \hat{\mathbf{b}}_i = \left(\hat{\boldsymbol{\gamma}} + \sum_{k=\hat{\kappa}+1}^{\kappa} \gamma^k \mathbf{p}_k \right)^T \hat{\mathbf{b}}_i \quad \text{for all } i \in I. \tag{15}$$

Consider any arbitrary $\boldsymbol{\alpha} = [\alpha^1, \alpha^2, ..., \alpha^\kappa]^T \geq \mathbf{0}$. According to (15), $\boldsymbol{\beta} = \left[\hat{\boldsymbol{\beta}}^T \, \mathbf{0}^T \right]^T \in \mathbb{R}^\kappa$ satisfies $S_1(\boldsymbol{\beta}) \subseteq S_1(\boldsymbol{\alpha})$, $S_0(\boldsymbol{\beta}) \subseteq S_0(\boldsymbol{\alpha})$, and $H(\boldsymbol{\beta}) \supseteq H(\boldsymbol{\alpha})$ if

$$\hat{\boldsymbol{\beta}} \in P(\boldsymbol{\alpha}) = \left\{ \hat{\boldsymbol{\beta}} \;\middle|\; \hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i \leq 1 \; \forall i \in S_1(\boldsymbol{\alpha}), \; \hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i \geq 1 \; \forall i \in S_0(\boldsymbol{\alpha}), \; \hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i = 1 \; \forall i \in H(\boldsymbol{\alpha}) \right\}.$$

Obviously, $\hat{\boldsymbol{\alpha}} + \sum_{k=\hat{\kappa}+1}^{\kappa} \alpha^k \mathbf{p}_k \in P(\boldsymbol{\alpha})$, where $\hat{\boldsymbol{\alpha}} = [\alpha^1, \alpha^2, ..., \alpha^{\hat{\kappa}}]^T$. Consider any $\hat{\boldsymbol{\beta}} \in P(\boldsymbol{\alpha})$ and $\mathbf{d} \in \mathbb{R}^{\hat{\kappa}}$ such that $\hat{\boldsymbol{\beta}} + \lambda \mathbf{d} \in P(\boldsymbol{\alpha})$ for all $\lambda \in \mathbb{R}$, i.e.,

$$\begin{aligned} \hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i + \lambda \mathbf{d}^T \hat{\mathbf{b}}_i / a_i &\leq 1 \quad \forall i \in S_1(\boldsymbol{\alpha}), \\ \hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i + \lambda \mathbf{d}^T \hat{\mathbf{b}}_i / a_i &\geq 1 \quad \forall i \in S_0(\boldsymbol{\alpha}), \\ \hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i + \lambda \mathbf{d}^T \hat{\mathbf{b}}_i / a_i &= 1 \quad \forall i \in H(\boldsymbol{\alpha}), \end{aligned}$$

for all $\lambda \in \mathbb{R}$. Thus, we have $\mathbf{d}^T \hat{\mathbf{b}}_i / a_i = 0$ for all $i \in I$, which yields $\mathbf{d} = \mathbf{0}$ as the set $\{\hat{\mathbf{b}}_i : i \in I\}$ contains $\hat{\kappa}$ linearly independent members. As a result, the polyhedron $P(\boldsymbol{\alpha})$ does not contain a line. Recall that $P(\boldsymbol{\alpha}) \neq \emptyset$. Therefore, it must have an extreme point. Let $\hat{\boldsymbol{\beta}} \in P(\boldsymbol{\alpha})$ be an extreme point of $P(\boldsymbol{\alpha})$. Among the constraints defining $P(\boldsymbol{\alpha})$, there must be $\hat{\kappa}$ linearly independent constraints tight at $\hat{\boldsymbol{\beta}}$. In other words, there exists $T \subseteq I$ such that $\hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i = 1$ for all $i \in T$, $|T| = \hat{\kappa}$, and $\{\hat{\mathbf{b}}_i / a_i : i \in T\}$ are linearly independent. Thus, Line 5 in Algorithm 1 considers some $\boldsymbol{\beta}$ such that $S_1(\boldsymbol{\beta}) \subseteq S_1(\boldsymbol{\alpha})$, $S_0(\boldsymbol{\beta}) \subseteq S_0(\boldsymbol{\alpha})$, and $H(\boldsymbol{\beta}) \supseteq H(\boldsymbol{\alpha})$.

Also note that for some $\boldsymbol{\beta}$ considered in Line 5, there may not exist $\boldsymbol{\alpha} \geq \mathbf{0}$ such that $S_1(\boldsymbol{\beta}) \subseteq S_1(\boldsymbol{\alpha})$, $S_0(\boldsymbol{\beta}) \subseteq S_0(\boldsymbol{\alpha})$, and $H(\boldsymbol{\beta}) \supseteq H(\boldsymbol{\alpha})$, i.e.,

$$P'(\boldsymbol{\beta}) = \left\{ \boldsymbol{\alpha} \geq \mathbf{0} \;\middle|\; \boldsymbol{\alpha}^T \mathbf{b}_i / a_i < 1 \; \forall i \in S_1(\boldsymbol{\beta}), \; \boldsymbol{\alpha}^T \mathbf{b}_i / a_i > 1 \; \forall i \in S_0(\boldsymbol{\beta}) \right\} = \emptyset.$$

Obviously, $P'(\boldsymbol{\beta}) = \emptyset$ if and only if the linear program $\mathcal{P}(\boldsymbol{\beta})$, defined in Line 8 of Algorithm 1, has a non-positive optimal value. Proposition 3 shows that there is no need to consider any $\boldsymbol{\beta}$ with $P'(\boldsymbol{\beta}) = \emptyset$, which explains Line 9 in Algorithm 1. □

A.7 Proof of Proposition 5

WLOG, suppose $i^* = 1$. Consider the following linear programming problem

$$\mathcal{P}_S : \quad \max \left\{ z_1 \;\middle|\; \sum_{i \in S} \hat{\mathbf{b}}_i z_i = \sum_{i \in S} \hat{\mathbf{b}}_i \bar{z}_i, \; z_i \in [0, 1], \; \forall i \in S \right\}$$

and the corresponding dual

$$\mathcal{D}_S: \quad \min \left\{ \boldsymbol{\theta}^T \left(\sum_{i \in S} \hat{\mathbf{b}}_i \bar{z}_i \right) + \sum_{i \in S} \vartheta_i \;\middle|\; \begin{array}{ll} \boldsymbol{\theta}^T \hat{\mathbf{b}}_1 + \vartheta_1 \geq 1 & \\ \boldsymbol{\theta}^T \hat{\mathbf{b}}_i + \vartheta_i \geq 0 & \forall i \in S \setminus \{1\} \\ \vartheta_i \geq 0 & \forall i \in S \end{array} \right\}.$$

Obviously, $(\bar{z}_i \, \forall i \in S)$ is a feasible solution to $\mathcal{P}_S$ and the feasible region of $\mathcal{P}_S$ is bounded. Therefore, there exists an optimal solution $(z_i^* \, \forall i \in S)$ to $\mathcal{P}_S$. If $z_1^* = 1$, we obtain a solution $(z_i^* \, \forall i \in S)$ satisfying the condition (i).

Suppose that $z_1^* < 1$. According to the complementary slackness conditions, there exists an optimal solution $(\boldsymbol{\theta}^*, \vartheta_i^* \, \forall i \in S)$ to $\mathcal{D}_S$ such that

$$(1 - z_1^*)\vartheta_1^* = 0, \quad z_1^*(\boldsymbol{\theta}^{*T} \hat{\mathbf{b}}_1 + \vartheta_1^* - 1) = 0, \quad (1 - z_i^*)\vartheta_i^* = 0, \quad \text{and} \quad z_i^*(\boldsymbol{\theta}^{*T} \hat{\mathbf{b}}_i + \vartheta_i^*) = 0 \;\; \forall i \in S \setminus \{1\}.$$

Therefore, $z_1^* < 1$ implies $\vartheta_1^* = 0$. The first constraint of $\mathcal{D}_S$ yields $\boldsymbol{\theta}^{*T} \hat{\mathbf{b}}_1 \geq 1 > 0$. For any $i \in S \setminus \{1\}$ and $\boldsymbol{\theta}^{*T} \hat{\mathbf{b}}_i > 0$, the second and third constraints of $\mathcal{D}_S$ yields $\boldsymbol{\theta}^{*T} \hat{\mathbf{b}}_i + \vartheta_i^* > 0$ and hence $z_i^* = 0$ by the complementary slackness condition. Similarly, for any $i \in S \setminus \{1\}$ and $\boldsymbol{\theta}^{*T} \hat{\mathbf{b}}_i < 0$, the second constraint of $\mathcal{D}_S$ implies $\vartheta_i^* > 0$ and $z_i^* = 1$ follows from the complementary slackness condition. As a result, we have $(z_i^* \, \forall i \in S)$ and $\boldsymbol{\theta}^*$ satisfying the condition (ii).

Now, consider the case that $\hat{\mathbf{b}}_1$ cannot be represented as a linear combination of $\{\hat{\mathbf{b}}_i : i \in S \backslash \{1\}\}$. Let $\tilde{\kappa}$ denote the number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in S\}$, i.e., $\tilde{\kappa} = \text{rank}(\mathbf{M}_S) \leq \hat{\kappa}$, where $\mathbf{M}_S$ is the $\hat{\kappa} \times |S|$ matrix each column of which corresponds to a distinct vector in $\{\hat{\mathbf{b}}_i : i \in S\}$. WLOG, assume the first $\tilde{\kappa}$ rows of $\mathbf{M}_S$ are linearly independent. Let $\tilde{\mathbf{b}}_i = [b_i^1, b_i^2, ..., b_i^{\tilde{\kappa}}]^T$ for all $i \in S$. Then for any $k \in \{\tilde{\kappa} + 1, ..., \hat{\kappa}\}$, there exists $\tilde{\mathbf{p}}_k \in \mathbb{R}^{\tilde{\kappa}}$ such that $b_i^k = \tilde{\mathbf{p}}_k^T \tilde{\mathbf{b}}_i$ for all $i \in S$. Furthermore, we have $\tilde{\kappa}$ linearly independent vectors in the set $\{\tilde{\mathbf{b}}_i : i \in S\}$.

Assume for contradiction that $\{\tilde{\mathbf{b}}_i : i \in S \setminus \{1\}\}$ have $\tilde{\kappa}$ linearly independent vectors. Then $\tilde{\mathbf{b}}_1$ can be presented as a linear combination of $\{\tilde{\mathbf{b}}_i : i \in S \setminus \{1\}\}$, i.e., there exist some $x_i \in \mathbb{R}$ for all $i \in S \setminus \{1\}$ such that $\tilde{\mathbf{b}}_1 = \sum_{i \in S \setminus \{1\}} \tilde{\mathbf{b}}_i x_i$. For any $k \in \{\tilde{\kappa} + 1, ..., \hat{\kappa}\}$, $b_i^k = \tilde{\mathbf{p}}_k^T \tilde{\mathbf{b}}_i$ for all $i \in S$ yields

$$b_1^k = \tilde{\mathbf{p}}_k^T \tilde{\mathbf{b}}_1 = \tilde{\mathbf{p}}_k^T \left(\sum_{i \in S \setminus \{1\}} \tilde{\mathbf{b}}_i x_i \right) = \sum_{i \in S \setminus \{1\}} \left(\tilde{\mathbf{p}}_k^T \tilde{\mathbf{b}}_i \right) x_i = \sum_{i \in S \setminus \{1\}} b_i^k x_i.$$

Therefore, $\hat{\mathbf{b}}_1$ can be represented as a linear combination of $\{\hat{\mathbf{b}}_i : i \in S \setminus \{1\}\}$, which results in a contradiction.

Thus, the number of linearly independent vectors in $\{\tilde{\mathbf{b}}_i : i \in S \setminus \{1\}\}$ must be strictly less than $\tilde{\kappa}$, which implies the existence of $\mathbf{x} \in \mathbb{R}^{\tilde{\kappa}} \setminus \{\mathbf{0}\}$ such that $\mathbf{x}^T \tilde{\mathbf{b}}_i = 0$ for all $i \in S \setminus \{1\}$. Recall that

$\{\tilde{\mathbf{b}}_i : i \in S\}$ contains $\tilde{\kappa}$ linearly independent vectors. Therefore, if $\mathbf{y}^T\tilde{\mathbf{b}}_i = 0$ for all $i \in S$, then $\mathbf{y} = \mathbf{0}$. As a result, we have $\mathbf{x}^T\tilde{\mathbf{b}}_1 \neq 0$. Let $\boldsymbol{\theta} = [\mathbf{x}^T\, \mathbf{0}^T]^T \in \mathbb{R}^{\hat{\kappa}}$ if $\mathbf{x}^T\tilde{\mathbf{b}}_1 > 0$ and $\boldsymbol{\theta} = [-\mathbf{x}^T\, \mathbf{0}^T]^T \in \mathbb{R}^{\hat{\kappa}}$ if $\mathbf{x}^T\tilde{\mathbf{b}}_1 < 0$. It is straightforward that $(\bar{z}_i\, \forall i \in S)$ and $\boldsymbol{\theta}$ satisfy the condition (ii). □

A.8 Proof of Proposition 6

We prove the result through the following four steps.

Step 1 shows that `solveP`$(F, U, \Psi(\cdot))$ is well-defined and terminates for any (F, U) satisfying the conditions (C1) and (C2). A sufficient condition for `solveP`$(F, U, \Psi(\cdot))$ to terminate is

(T) $|\bar{U}| < |U|$ for any $\bar{U}$ used in Line 7 and $|\bar{U}(\boldsymbol{\theta})| < |U|$ for any $\bar{U}(\boldsymbol{\theta})$ used in Line 15.

As a majority of lines in Algorithm 2 are straightforward, we focus on the following lines:

Line 2. The condition (C1) asserts that $\{\hat{\mathbf{b}}_i : i \in F\}$ have $|F|$ linearly independent vectors, which implies $|F| \leq \hat{\kappa}$. If $|F| = \hat{\kappa}$, according to the condition (C1), there is no linearly independent vector in $\{\hat{\mathbf{b}}_i : i \in U\}$, and hence $U = \emptyset$. As a result, Lines 3 to 18 are executed only when $|F| < \hat{\kappa}$.

Line 7. Note that $\bar{U} \subset U$. It is straightforward that $(F, \bar{U})$ satisfies the condition (C2). Since $\hat{\mathbf{b}}_{i^*}$ can be represented as a linear combination of $\{\hat{\mathbf{b}}_i : i \in U \setminus \{i^*\}\}$, the number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in U \setminus \{i^*\}\}$ is the same as that in $\{\hat{\mathbf{b}}_i : i \in U\}$, which equals to $\hat{\kappa} - |F|$ by the condition (C1) for (F, U). As $\bar{U} = U \setminus \{i^*\}$, $(F, \bar{U})$ also satisfies the condition (C1). Therefore, it is legitimate to call the function `solveP`$(F, \bar{U}, \bar{\Psi}(\cdot))$. Furthermore, the condition (T) is also satisfied as $|\bar{U}| = |U| - 1$.

Line 11. Consider $x_i \in \mathbb{R}$ for all $i \in F \cup \{i^*\} \cup U_f$ such that

$$\sum_{i \in F} \hat{\mathbf{b}}_i x_i + \hat{\mathbf{b}}_{i^*} x_{i^*} + \sum_{i \in U_f} \hat{\mathbf{b}}_i x_i = \mathbf{0}.$$

Note that $\mathbf{c}_F \equiv \sum_{i \in F} \hat{\mathbf{b}}_i x_i$ and $\mathbf{c}_U \equiv \hat{\mathbf{b}}_{i^*} x_{i^*} + \sum_{i \in U_f} \hat{\mathbf{b}}_i x_i$ are linear combinations of $\{\hat{\mathbf{b}}_i : i \in F\}$ and $\{\hat{\mathbf{b}}_i : i \in U\}$, respectively. According to (C2), if $\mathbf{c}_F \neq \mathbf{0}$ and $\mathbf{c}_U \neq \mathbf{0}$, $\mathbf{c}_F$ and $\mathbf{c}_U$ are linearly independent and hence it is impossible to obtain $\mathbf{c}_F + \mathbf{c}_U = \mathbf{0}$. Therefore, we have $\mathbf{c}_F = \mathbf{c}_U = \mathbf{0}$. As the set $\{\hat{\mathbf{b}}_i : i \in F\}$ has $|F|$ independent vectors, $\mathbf{c}_F = \mathbf{0}$ implies $x_i = 0$ for all $i \in F$. Note that $\mathbf{c}_U$ can also be viewed as a linear combination of the set $\{\hat{\mathbf{b}}_i : i \in U_f \cup \{i^*\}\}$, which, according to Line 10, contains $\hat{\kappa} - |F| = |U_f| + 1$ linearly independent vectors. Consequently, $\mathbf{c}_U = \mathbf{0}$ implies $x_i = 0$ for all $i \in U_f \cup \{i^*\}$. Therefore, the vectors in $\{\hat{\mathbf{b}}_i : i \in F \cup \{i^*\} \cup U_f\}$ are all linearly independent, and hence $\boldsymbol{\theta}$ in Line 11 is well-defined.

Line 15. The following arguments establish that $(\bar{F}, \bar{U}(\boldsymbol{\theta}))$ satisfies the condition (C1):

- The analysis for Line 11 shows that the vectors in $\{\hat{\mathbf{b}}_i : i \in F \cup \{i^*\} \cup U_f\}$ are all linearly independent. Therefore, there are $|F| + 1$ linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in \bar{F}\}$.

- The analysis for Line 11 also yields that the set $\{\hat{\mathbf{b}}_i : i \in F \cup \{i^*\} \cup U_f\}$ includes $\hat{\kappa}$ linearly independent vectors. Thus, for any $\iota \in \bar{U}(\boldsymbol{\theta}) \backslash U_f$, $\hat{\mathbf{b}}_\iota$ can be represented as a linear combination of $\{\hat{\mathbf{b}}_i : i \in F \cup \{i^*\} \cup U_f\}$, i.e., there exist $x_i \in \mathbb{R}$ for all $i \in F \cup \{i^*\} \cup U_f$ such that

$$\sum_{i \in F} \hat{\mathbf{b}}_i x_i + \sum_{i \in U_f \cup \{i^*\}} \hat{\mathbf{b}}_i x_i - \hat{\mathbf{b}}_\iota = \mathbf{0}.$$

  $\mathbf{c}_F \equiv \sum_{i \in F} \hat{\mathbf{b}}_i x_i$ and $\mathbf{c}_U \equiv \sum_{i \in U_f \cup \{i^*\}} \hat{\mathbf{b}}_i x_i - \hat{\mathbf{b}}_\iota$ are linear combinations of $\{\hat{\mathbf{b}}_i : i \in F\}$ and $\{\hat{\mathbf{b}}_i : i \in U\}$, respectively. Applying the condition (C2) for (F, U), $\mathbf{c}_F + \mathbf{c}_U = \mathbf{0}$ yields $\mathbf{c}_U = \mathbf{0}$, i.e., $\hat{\mathbf{b}}_\iota = \sum_{i \in U_f \cup \{i^*\}} \hat{\mathbf{b}}_i x_i$ and hence

$$\boldsymbol{\theta}^T \hat{\mathbf{b}}_\iota = \boldsymbol{\theta}^T \left(\sum_{i \in U_f \cup \{i^*\}} \hat{\mathbf{b}}_i x_i \right) = \sum_{i \in U_f \cup \{i^*\}} \boldsymbol{\theta}^T \hat{\mathbf{b}}_i x_i. \tag{16}$$

  According to the definitions of $\boldsymbol{\theta}$ and $\bar{U}(\boldsymbol{\theta})$ in Lines 11 and 13, $\boldsymbol{\theta}^T \hat{\mathbf{b}}_\iota = 0$, $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0$ for all $i \in U_f$, and $\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1$. Therefore, (16) yields $x_{i^*} = 0$ and so $\hat{\mathbf{b}}_\iota = \sum_{i \in U_f} \hat{\mathbf{b}}_i x_i$, which implies the number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in \bar{U}(\boldsymbol{\theta})\}$ is equal to that in $\{\hat{\mathbf{b}}_i : i \in U_f\}$. Recall that all the vectors in $\{\hat{\mathbf{b}}_i : i \in U_f\}$ are linearly independent. Hence, the set $\{\hat{\mathbf{b}}_i : i \in \bar{U}(\boldsymbol{\theta})\}$ contains $|U_f| = \hat{\kappa} - |F| - 1$ linearly independent vectors.

To show that $(\bar{F}, \bar{U}(\boldsymbol{\theta}))$ satisfies (C2), assume for contradiction that there exist linearly dependent $\mathbf{c}_{\bar{F}}$ and $\mathbf{c}_{\bar{U}}$, which are non-zero linear combinations of $\{\hat{\mathbf{b}}_i : i \in \bar{F}\}$ and $\{\hat{\mathbf{b}}_i : i \in \bar{U}(\boldsymbol{\theta})\}$, respectively. Let $\mathbf{c}_{\bar{F}} \equiv \sum_{i \in \bar{F}} \hat{\mathbf{b}}_i x_i$ where $x_i \in \mathbb{R}$ for all $i \in \bar{F} = F \cup \{i^*\}$ and $\mathbf{c}_{\bar{U}} \equiv \sum_{i \in \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i x_i$ where $x_i \in \mathbb{R}$ for all $i \in \bar{U}(\boldsymbol{\theta})$. As $\mathbf{c}_{\bar{F}}$ and $\mathbf{c}_{\bar{U}}$ are linearly dependent, there exists some $\lambda \neq 0$ such that

$$\mathbf{0} = \mathbf{c}_{\bar{F}} + \lambda \mathbf{c}_{\bar{U}} = \sum_{i \in \bar{F}} \hat{\mathbf{b}}_i x_i + \lambda \sum_{i \in \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i x_i = \sum_{i \in F} \hat{\mathbf{b}}_i x_i + \hat{\mathbf{b}}_{i^*} x_{i^*} + \lambda \sum_{i \in \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i x_i = \mathbf{d}_F + \mathbf{d}_U,$$

where $\mathbf{d}_F \equiv \sum_{i \in F} \hat{\mathbf{b}}_i x_i$ and $\mathbf{d}_U \equiv \hat{\mathbf{b}}_{i^*} x_{i^*} + \lambda \sum_{i \in \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i x_i$. According to the condition (C2) for (F, U), $\mathbf{d}_F$ and $\mathbf{d}_U$ are linearly independent as long as $\mathbf{d}_F \neq \mathbf{0}$ and $\mathbf{d}_U \neq \mathbf{0}$. Therefore, $\mathbf{d}_F + \mathbf{d}_U = \mathbf{0}$ implies $\mathbf{d}_F = \mathbf{d}_U = \mathbf{0}$. The definitions of $\mathbf{c}_{\bar{F}}$ and $\mathbf{d}_F$ yields

$$\mathbf{c}_{\bar{F}} = \mathbf{d}_F + \hat{\mathbf{b}}_{i^*} x_{i^*} = \hat{\mathbf{b}}_{i^*} x_{i^*},$$

which implies $x_{i^*} \neq 0$ as $\mathbf{c}_{\bar{F}} \neq \mathbf{0}$. Applying $\mathbf{d}_U = \mathbf{0}$, we obtain $\hat{\mathbf{b}}_{i^*} = -\frac{\lambda}{x_{i^*}} \sum_{i \in \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i x_i$. Recall that $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0$ for all $i \in \bar{U}(\boldsymbol{\theta})$. Thus,

$$\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = \boldsymbol{\theta}^T \left(-\frac{\lambda}{x_{i^*}} \sum_{i \in \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i x_i \right) = -\frac{\lambda}{x_{i^*}} \sum_{i \in \bar{U}(\boldsymbol{\theta})} \boldsymbol{\theta}^T \hat{\mathbf{b}}_i x_i = 0,$$

which contradicts $\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1$ in Line 11.

To sum up, $(\bar{F}, \bar{U}(\boldsymbol{\theta}))$ satisfies the conditions (C1) and (C2), so it is legitimate to call the function $\texttt{solveP}(\bar{F}, \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$ in Line 15. Furthermore, we have $\bar{U}(\boldsymbol{\theta}) \subseteq U \setminus \{i^*\}$ and hence $|\bar{U}(\boldsymbol{\theta})| \leq |U| - 1$, which satisfies the condition (T).

Step 2 shows that the function $\texttt{solveP}(F, U, \Psi(\cdot))$ solves $\mathcal{P}(F \cup U, \Psi(\cdot))$ for any (F, U) satisfying the conditions (C1) and (C2). According to Line 2, this statement is trivially true when $|U| = 0$. Assume that the statement holds for any (F, U) satisfying (C1), (C2), and $|U| \leq \mu$, where μ is a nonnegative integer less than $|I|$. By mathematical induction, it suffices to prove the statement when $|U| = \mu + 1$.

Let $(z_i^* \, \forall i \in F \cup U)$ be an optimal solution to $\mathcal{P}(F \cup U, \Psi(\cdot))$. Consider $i^* \in U$ picked in Line 4. According to Proposition 5, it is sufficient to consider the following two cases:

Case 1 in Step 2. Suppose that $\hat{\mathbf{b}}_{i^*}$ can be represented as a linear combination of $\{\hat{\mathbf{b}}_i : i \in F \cup U \setminus \{i^*\}\}$ and there exist $\tilde{z}_i \in [0, 1]$ for all $i \in F \cup U$ with $\sum_{i \in F \cup U} \hat{\mathbf{b}}_i \tilde{z}_i = \sum_{i \in F \cup U} \hat{\mathbf{b}}_i z_i^*$ and satisfying the condition (i) in Proposition 5, i.e., $\tilde{z}_{i^*} = 1$.

As $\hat{\mathbf{b}}_{i^*}$ is a linear combination of $\{\hat{\mathbf{b}}_i : i \in F \cup U \setminus \{i^*\}\}$, there exist $x_i \in \mathbb{R}$ for all $i \in F \cup U \setminus \{i^*\}$ such that

$$\sum_{i \in F} \hat{\mathbf{b}}_i x_i + \sum_{i \in U \setminus \{i^*\}} \hat{\mathbf{b}}_i x_i - \hat{\mathbf{b}}_{i^*} = \mathbf{0}.$$

The condition (C2) for (F, U) yields $\sum_{i \in U \setminus \{i^*\}} \hat{\mathbf{b}}_i x_i - \hat{\mathbf{b}}_{i^*} = \mathbf{0}$, i.e., $\hat{\mathbf{b}}_{i^*}$ can be represented as a linear combination of $\{\hat{\mathbf{b}}_i : i \in U \setminus \{i^*\}\}$. Therefore, the condition in Line 5 is satisfied.

According to the induction assumption, Line 7 finds an optimal solution $(\bar{z}_i \, \forall i \in F \cup U \setminus \{i^*\})$ to the problem

$$\mathcal{P}(F \cup \bar{U}, \bar{\Psi}(\cdot)) : \quad \min \left\{ \Psi \left(\hat{\mathbf{b}}_{i^*} + \sum_{i \in F \cup U \setminus \{i^*\}} \hat{\mathbf{b}}_i z_i \right) \;\middle|\; z_i \in [0, 1] \; \forall i \in F \cup U \setminus \{i^*\} \right\}.$$

Note that $(\tilde{z}_i \, \forall i \in F \cup U \setminus \{i^*\})$ is a feasible solution to $\mathcal{P}(F \cup \bar{U}, \bar{\Psi}(\cdot))$. Therefore, we obtain

$$\Psi \left(\hat{\mathbf{b}}_{i^*} + \sum_{i \in F \cup U \setminus \{i^*\}} \hat{\mathbf{b}}_i \bar{z}_i \right) \leq \Psi \left(\hat{\mathbf{b}}_{i^*} + \sum_{i \in F \cup U \setminus \{i^*\}} \hat{\mathbf{b}}_i \tilde{z}_i \right) = \Psi \left(\sum_{i \in F \cup U} \hat{\mathbf{b}}_i \tilde{z}_i \right) = \Psi \left(\sum_{i \in F \cup U} \hat{\mathbf{b}}_i z_i^* \right), \tag{17}$$

where the two equalities follow from $\tilde{z}_{i^*} = 1$ and $\sum_{i \in F \cup U} \hat{\mathbf{b}}_i \tilde{z}_i = \sum_{i \in F \cup U} \hat{\mathbf{b}}_i z_i^*$, respectively.

Let $\bar{z}_{i^*} = 1$. Consider the problem $\mathcal{P}(F \cup U, \Psi(\cdot))$ in (8). Obviously, $(\bar{z}_i \, \forall i \in F \cup U)$ is feasible to $\mathcal{P}(F \cup U, \Psi(\cdot))$. As $(z_i^* \, \forall i \in F \cup U)$ is its optimal solution, (17) implies that $(\bar{z}_i \, \forall i \in F \cup U)$ is also optimal to $\mathcal{P}(F \cup U, \Psi(\cdot))$. Line 8 suggests that $\texttt{solveP}(F, U, \Psi(\cdot))$ solves $\mathcal{P}(F \cup U, \Psi(\cdot))$.

Case 2 in Step 2. Suppose that there exist $\tilde{z}_i \in [0, 1]$ for all $i \in F \cup U$ with $\sum_{i \in F \cup U} \hat{\mathbf{b}}_i \tilde{z}_i = \sum_{i \in F \cup U} \hat{\mathbf{b}}_i z_i^*$ and satisfying the condition (ii) in Proposition 5, i.e., we can obtain some $\boldsymbol{\vartheta}$ such that $\boldsymbol{\vartheta}^T \hat{\mathbf{b}}_{i^*} > 0$, $\tilde{z}_i = 0$ for all $i \in F \cup U \setminus \{i^*\}$ and $\boldsymbol{\vartheta}^T \hat{\mathbf{b}}_i > 0$, and $\tilde{z}_i = 1$ for all $i \in F \cup U \setminus \{i^*\}$ and $\boldsymbol{\vartheta}^T \hat{\mathbf{b}}_i < 0$.

According to the condition (C1), we can choose $U_{ind} \subseteq U$ and $|U_{ind}| = \hat{\kappa} - |F|$ such that $\{\hat{\mathbf{b}}_i : i \in U_{ind}\}$ are linearly independent. Consider $x_i \in \mathbb{R}$ for all $i \in F \cup U_{ind}$ such that

$$\sum_{i \in F} \hat{\mathbf{b}}_i x_i + \sum_{i \in U_{ind}} \hat{\mathbf{b}}_i x_i = \mathbf{0}.$$

The condition (C2) for (F, U) yields $\sum_{i \in F} \hat{\mathbf{b}}_i x_i = \sum_{i \in U_{ind}} \hat{\mathbf{b}}_i x_i = \mathbf{0}$. The condition (C1) implies all the vectors in $\{\hat{\mathbf{b}}_i : i \in F\}$ are linearly independent. Also note that $\{\hat{\mathbf{b}}_i : i \in U_{ind}\}$ are linearly independent. Therefore, we have $x_i = 0$ for all $i \in F \cup U_{ind}$, i.e., $\{\hat{\mathbf{b}}_i : i \in F \cup U_{ind}\}$ are linearly independent. The condition (C2) also implies $F \cap U = \emptyset$ and hence $|F \cup U_{ind}| = \hat{\kappa}$. As a result, there are $\hat{\kappa}$ linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in F \cup U\}$.

To simplify the notation, let $T_0 \equiv \{i \in F \cup U \setminus \{i^*\} : \boldsymbol{\vartheta}^T \hat{\mathbf{b}}_i > 0\}$, $T_1 \equiv \{i \in F \cup U \setminus \{i^*\} : \boldsymbol{\vartheta}^T \hat{\mathbf{b}}_i < 0\}$, and $T_f \equiv F \cup U \setminus \{i^*\} \setminus T_0 \setminus T_1$. Define

$$P \equiv \left\{ \boldsymbol{\theta} \;\middle|\; \begin{array}{ll} \boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1, & \boldsymbol{\theta}^T \hat{\mathbf{b}}_i \geq 0 \quad \forall i \in T_0, \\ \boldsymbol{\theta}^T \hat{\mathbf{b}}_i \leq 0 \quad \forall i \in T_1, & \boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0 \quad \forall i \in T_f \end{array} \right\}.$$

Consider any $\boldsymbol{\theta} \in P$ and $\mathbf{d} \in \mathbb{R}^{\hat{\kappa}}$ such that $\boldsymbol{\theta} + \lambda \mathbf{d} \in P$ for all $\lambda \in \mathbb{R}$, i.e.,

$$\begin{array}{ll} \boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} + \lambda \mathbf{d}^T \hat{\mathbf{b}}_{i^*} = 1, & \boldsymbol{\theta}^T \hat{\mathbf{b}}_i + \lambda \mathbf{d}^T \hat{\mathbf{b}}_i \geq 0 \quad \forall i \in T_0, \\ \boldsymbol{\theta}^T \hat{\mathbf{b}}_i + \lambda \mathbf{d}^T \hat{\mathbf{b}}_i \leq 0 \quad \forall i \in T_1, & \boldsymbol{\theta}^T \hat{\mathbf{b}}_i + \lambda \mathbf{d}^T \hat{\mathbf{b}}_i = 0 \quad \forall i \in T_f, \end{array}$$

for all $\lambda \in \mathbb{R}$. It immediately implies $\mathbf{d}^T \hat{\mathbf{b}}_i = 0$ for all $i \in \{i^*\} \cup T_0 \cup T_1 \cup T_f = F \cup U$. Recall that there are $\hat{\kappa}$ linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in F \cup U\}$. We obtain $\mathbf{d} = \mathbf{0}$. Also note that $\boldsymbol{\vartheta} / (\boldsymbol{\vartheta}^T \hat{\mathbf{b}}_{i^*}) \in P$. The polyhedron P must has an extreme point.

Consider an extreme point $\bar{\boldsymbol{\theta}}$ of P. Define $T \subseteq F \cup U \setminus \{i^*\}$ such that the tight constraints at $\bar{\boldsymbol{\theta}}$ are $\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1$ and $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0$ for all $i \in T$. Note that there are $\hat{\kappa}$ linearly independent tight constraints at $\bar{\boldsymbol{\theta}}$. The set $\{\hat{\mathbf{b}}_i : i \in T \cup \{i^*\}\}$ must have $\hat{\kappa}$ linearly independent vectors. Assume for contradiction that there exists some $\iota \in F \setminus T$. Then the number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in T \cup \{i^*\}\}$ is no greater than that of $\{\hat{\mathbf{b}}_i : i \in F \cup U \setminus \{\iota\}\}$. The number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in F \cup U \setminus \{\iota\}\}$ is less than the sum of (a) the number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in F \setminus \{\iota\}\}$ and (b) the number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in U\}$. The term (a) is at most $|F \setminus \{\iota\}| = |F| - 1$, while the term (b) is $\hat{\kappa} - |F|$ by the condition (C1). Consequently, the number of linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in T \cup \{i^*\}\}$ is no greater than $\hat{\kappa} - 1$, which results in a contradiction.

The analysis in the previous paragraph indicates that the constraints $\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1$ and $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0$ for all $i \in F$ are tight at any extreme point of P. Combining with the analysis for Line 11 in Step 1, we obtain that any extreme point of P must be considered in Line 13.

Let $\boldsymbol{\theta}$ denote an extreme point of P considered in Line 13. Consider the problem $\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$ solved in Line 15:

$$\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot)) : \quad \min \left\{ \Psi \left(\sum_{i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i z_i + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0} \hat{\mathbf{b}}_i \right) \;\middle|\; z_i \in [0,1] \; \forall i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta}) \right\}.$$

Note that $(\tilde{z}_i \, \forall i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta}))$ is feasible to $\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$ and the corresponding objective function value is

$$\Psi \left(\sum_{i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i \tilde{z}_i + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0} \hat{\mathbf{b}}_i \right) = \Psi \left(\sum_{i \in F \cup \{i^*\}} \hat{\mathbf{b}}_i \tilde{z}_i + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0} \hat{\mathbf{b}}_i \tilde{z}_i + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0} \hat{\mathbf{b}}_i \right).$$

For any $i \in U$ such that $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0$, the definitions of P and T_1 yield $i \in T_1 = \{i \in F \cup U \setminus \{i^*\} : \boldsymbol{\vartheta}^T \hat{\mathbf{b}}_i < 0\}$, which implies $\tilde{z}_i = 1$ by the condition (ii). Similarly, for any $i \in U \setminus \{i^*\}$ such that $\boldsymbol{\theta}^T \hat{\mathbf{b}}_i > 0$, the definitions of P and T_0 yield $i \in T_0 = \{i \in F \cup U \setminus \{i^*\} : \boldsymbol{\vartheta}^T \hat{\mathbf{b}}_i > 0\}$, which implies $\tilde{z}_i = 0$. Therefore, we obtain

$$\Psi \left(\sum_{i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i \tilde{z}_i + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0} \hat{\mathbf{b}}_i \right) = \Psi \left(\sum_{i \in F \cup U} \hat{\mathbf{b}}_i \tilde{z}_i \right) = \Psi \left(\sum_{i \in F \cup U} \hat{\mathbf{b}}_i z_i^* \right),$$

where the last equality follows from $\sum_{i \in F \cup U} \hat{\mathbf{b}}_i \tilde{z}_i = \sum_{i \in F \cup U} \hat{\mathbf{b}}_i z_i^*$.

Now consider the solution $(\bar{z}_i \, \forall i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta}))$ obtained in Line 15. Note that $\bar{U}(\boldsymbol{\theta}) \subseteq U$ and $i^* \in U \setminus \bar{U}(\boldsymbol{\theta})$ as $\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1$, implying $|\bar{U}(\boldsymbol{\theta})| \le |U| - 1 \le \mu$. By the induction assumption, $(\bar{z}_i \, \forall i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta}))$ is optimal to $\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$. We have

$$\Psi \left(\sum_{i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i \bar{z}_i + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0} \hat{\mathbf{b}}_i \right) \le \Psi \left(\sum_{i \in \bar{F} \cup \bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i \tilde{z}_i + \sum_{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0} \hat{\mathbf{b}}_i \right) = \Psi \left(\sum_{i \in F \cup U} \hat{\mathbf{b}}_i z_i^* \right). \quad (18)$$

Set $\bar{z}_i = 1$ for all $\{i \in U : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i < 0\}$ and $\bar{z}_i = 0$ for all $\{i \in U \setminus \{i^*\} : \boldsymbol{\theta}^T \hat{\mathbf{b}}_i > 0\}$. We obtain

$$\Psi\left(\sum_{\bar{F}\cup\bar{U}(\boldsymbol{\theta})} \hat{\mathbf{b}}_i \bar{z}_i + \sum_{i\in U:\boldsymbol{\theta}^T\hat{\mathbf{b}}_i<0} \hat{\mathbf{b}}_i\right) = \Psi\left(\sum_{i\in F\cup U} \hat{\mathbf{b}}_i \bar{z}_i\right).$$

Obviously, $(\bar{z}_i \, \forall i \in F \cup U)$ is feasible to $\mathcal{P}(F \cup U, \Psi(\cdot))$. (18) and the optimality of $(z_i^* \, \forall i \in F \cup U)$ yield that $(\bar{z}_i \, \forall i \in F \cup U)$ is also optimal to $\mathcal{P}(F \cup U, \Psi(\cdot))$. Line 16 ensures that we solve $\mathcal{P}(F \cup U, \Psi(\cdot))$ to optimality.

Step 3 shows that `solveP`$(F, U, \Psi(\cdot))$ decomposes the problem $\mathcal{P}(F\cup U, \Psi(\cdot))$ into a sequence of $\hat{\kappa}$-variable problems in the form of $\mathcal{P}(S, \Psi(\cdot))$, where S is distinct for all the decomposed problems.

The function `solveP`$(F, U, \Psi(\cdot))$ solves the problem $\mathcal{P}(F \cup U, \Psi(\cdot))$ by recursively comparing the optimal values of the problems solved in Line 2 of Algorithm 2. Formally, let $\omega_{F,U}$ denote the total number of the problems in Line 2 solved when executing the function `solveP`$(F, U, \Psi(\cdot))$. The wth problem solved is denoted by $\mathcal{P}(S^w_{F,U}, \Psi^w_{F,U}(\cdot))$. The sequence of the problems solved when executing `solveP`$(F, U, \Psi(\cdot))$ is then denoted by

$$\mathcal{S}_{F,U} = \left[\mathcal{P}(S^1_{F,U}, \Psi^1_{F,U}(\cdot)), ..., \mathcal{P}(S^{\omega_{F,U}}_{F,U}, \Psi^{\omega_{F,U}}_{F,U}(\cdot))\right]. \tag{19}$$

Note that Line 2 is executed only when the second input argument of the function `solveP`$(\cdot, \cdot, \cdot)$ is an empty set. Therefore, for any $w \in \{1, ..., \omega_{F,U}\}$, we solve the problem $\mathcal{P}(S^w_{F,U}, \Psi^w_{F,U}(\cdot))$ in Line 2 by calling `solveP`$(S^w_{F,U}, \emptyset, \Psi^w_{F,U}(\cdot))$. According to the condition (C1), we have $|S^w_{F,U}| = \hat{\kappa}$, i.e., $\mathcal{P}(S^w_{F,U}, \Psi^w_{F,U}(\cdot))$ has $\hat{\kappa}$ decision variables. Furthermore, the condition (C1) also implies that there are $\hat{\kappa}$ linearly independent vectors in the set $\{\hat{\mathbf{b}}_i : i \in S^w_{F,U}\}$.

Next, we would like to show that

(P) $S^w_{F,U} \neq S^{w'}_{F,U}$ for any $w \neq w'$ and $w, w' \in \{1, ..., \omega_{F,U}\}$.

If $U = \emptyset$, then we only solve one problem in Line 2, i.e., $\mathcal{P}(F, \Psi(\cdot))$, so the property (P) is trivially true. Assume for induction that the property (P) holds for all given $F, U \subseteq I$ with $|U| \leq \mu \in \{0, 1, ..., |I| - 1\}$ and satisfying the conditions (C1) and (C2). Consider an arbitrary $F, U \subseteq I$ with $|U| = \mu + 1$ and satisfying (C1) and (C2). Note that the sequence $\mathcal{S}_{F,U}$ can be partitioned into the subsequences of

(a) $\mathcal{S}_{F,\bar{U}}$ for the problem $\mathcal{P}(F \cup \bar{U}, \bar{\Psi}(\cdot))$ solved in Line 7;

(b) $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta})}$ for the problem $\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$ solved in Line 15 corresponding to any $\boldsymbol{\theta}$ considered in Line 13.

As $|\bar{U}| < |U|$ and $|\bar{U}(\boldsymbol{\theta})| < |U|$ for any $\bar{U}$ and $\bar{U}(\boldsymbol{\theta})$ used in Lines 7 and 15, the induction assumption immediately yields that any subsequence $\mathcal{S}_{F,\bar{U}}$ or $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta})}$ in parts (a) or (b) satisfies the property (P). It is sufficient to consider the following two cases:

Case 1 in Step 3 shows that $S^{w}_{F,U} \neq S^{w'}_{F,U}$ for any w and w' such that $\mathcal{P}(S^{w}_{F,U}, \Psi^{w}_{F,U}(\cdot))$ and $\mathcal{P}(S^{w'}_{F,U}, \Psi^{w'}_{F,U}(\cdot))$ belong to the subsequences $\mathcal{S}_{F,\bar{U}}$ in part (a) and $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta})}$ in part (b), respectively.

The subsequence $\mathcal{S}_{F,\bar{U}}$ in part (a) corresponds to the problem $\mathcal{P}(F \cup \bar{U}, \bar{\Psi}(\cdot))$ solved in Line 7. Note that $i^* \notin F \cup \bar{U}$. Hence, we have $i^* \notin S^{w}_{F,U}$ for any $\mathcal{P}(S^{w}_{F,U}, \Psi^{w}_{F,U}(\cdot))$ in the subsequence $\mathcal{S}_{F,\bar{U}}$. On the other hand, the subsequence $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta})}$ in part (b) corresponds to the problem $\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$ solved in Line 15. It is straightforward that $i^* \in \bar{F} \subseteq S^{w'}_{F,U}$ for any $\mathcal{P}(S^{w'}_{F,U}, \Psi^{w'}_{F,U}(\cdot))$ in the subsequence $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta})}$. As a result, we obtain $S^{w}_{F,U} \neq S^{w'}_{F,U}$.

Case 2 in Step 3 shows that $S^{w}_{F,U} \neq S^{w'}_{F,U}$ for any w and w' such that $\mathcal{P}(S^{w}_{F,U}, \Psi^{w}_{F,U}(\cdot))$ and $\mathcal{P}(S^{w'}_{F,U}, \Psi^{w'}_{F,U}(\cdot))$ belong to the subsequences $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta})}$ and $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta}')}$ in part (b), respectively, where $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$ are two distinct vectors generated by Line 11 in different iterations.

Consider $\mathcal{P}(S^{w}_{F,U}, \Psi^{w}_{F,U}(\cdot))$ in the subsequence $\mathcal{S}_{\bar{F},\bar{U}(\boldsymbol{\theta})}$, which corresponds to the problem $\mathcal{P}(\bar{F} \cup \bar{U}(\boldsymbol{\theta}), \bar{\Psi}(\boldsymbol{\theta}, \cdot))$ solved in Line 15. We have $i^* \in S^{w}_{F,U} \subseteq (F \cup \{i^*\} \cup \bar{U}(\boldsymbol{\theta}))$. According to the definitions of $\boldsymbol{\theta}$ and $\bar{U}(\boldsymbol{\theta})$ in Lines 11 and 13, we obtain

$$\boldsymbol{\theta}^T \hat{\mathbf{b}}_{i^*} = 1 \quad \text{and} \quad \boldsymbol{\theta}^T \hat{\mathbf{b}}_i = 0 \text{ for all } i \in S^{w}_{F,U} \setminus \{i^*\}. \tag{20}$$

Recall that there are $\hat{\kappa}$ linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in S^{w}_{F,U}\}$. Therefore, $\boldsymbol{\theta}$ is the unique solution to the linear equations in (20).

Symmetrically, we have $i^* \in S^{w'}_{F,U} \subseteq (F \cup \{i^*\} \cup \bar{U}(\boldsymbol{\theta}'))$ and $\boldsymbol{\theta}'$ is the unique solution to the linear equations $\boldsymbol{\theta}'^T \hat{\mathbf{b}}_{i^*} = 1$ and $\boldsymbol{\theta}'^T \hat{\mathbf{b}}_i = 0$ for all $i \in S^{w'}_{F,U} \setminus \{i^*\}$. Therefore, $\boldsymbol{\theta} \neq \boldsymbol{\theta}'$ implies $S^{w}_{F,U} \neq S^{w'}_{F,U}$.

Step 4 completes the proof by establishing the legitimacy of calling $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$. The condition (C2) holds trivially as the first input argument is an empty set. For any $\boldsymbol{\beta}$ constructed in Line 3 of Algorithm 1, there exist $\hat{\boldsymbol{\beta}} \in \mathbb{R}^{\hat{\kappa}}$ and $T \subseteq I$ such that $\boldsymbol{\beta} = \left[\hat{\boldsymbol{\beta}}^T \, \mathbf{0}^T\right]^T$, $\hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i = 1$ for all $i \in T$, $|T| = \hat{\kappa}$, and $\{\hat{\mathbf{b}}_i : i \in T\}$ are linearly independent. Applying the definition of $H(\boldsymbol{\beta})$ in (6), it is straightforward that $T \subseteq H(\boldsymbol{\beta})$. Therefore, there are $\hat{\kappa}$ linearly independent vectors in $\{\hat{\mathbf{b}}_i : i \in H(\boldsymbol{\beta})\}$, which satisfies the condition (C1).

According to Step 2, $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$ returns the optimal solution of $\mathcal{P}(H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$, which is equivalent to $\mathcal{R}(\boldsymbol{\beta})$. Step 3 immediately yields that $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$ solves at most $C^{\hat{\kappa}}_{|H(\boldsymbol{\beta})|}$ problems with $\hat{\kappa}$ variables in the form $\mathcal{P}(S, \Psi(\cdot))$. □

A.9 Proof of Theorem 3

Consider $\hat{\boldsymbol{\beta}} \neq \hat{\boldsymbol{\beta}}'$ generated by Line 3 in two iterations of Algorithm 1. We have $\boldsymbol{\beta} \neq \boldsymbol{\beta}'$ where $\boldsymbol{\beta} = \left[\hat{\boldsymbol{\beta}}^T \, \mathbf{0}^T\right]^T$ and $\boldsymbol{\beta}' = \left[\hat{\boldsymbol{\beta}}'^T \, \mathbf{0}^T\right]^T$. The corresponding problems $\mathcal{R}(\boldsymbol{\beta})$ and $\mathcal{R}(\boldsymbol{\beta}')$ in Line 11 are equivalent to $\mathcal{P}(H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$ and $\mathcal{P}(H(\boldsymbol{\beta}'), \hat{\Phi}(\boldsymbol{\beta}', \cdot))$, respectively, which can be solved by $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$ and $\texttt{solveP}(\emptyset, H(\boldsymbol{\beta}'), \hat{\Phi}(\boldsymbol{\beta}', \cdot))$ in Algorithm 2. Adopting the notations in (19), $\mathcal{P}(H(\boldsymbol{\beta}), \hat{\Phi}(\boldsymbol{\beta}, \cdot))$ and $\mathcal{P}(H(\boldsymbol{\beta}'), \hat{\Phi}(\boldsymbol{\beta}', \cdot))$ are decomposed into the problems in the sequences

$$
\begin{aligned}
\mathcal{S}_{\emptyset,H(\boldsymbol{\beta})} &= \left[\mathcal{P}\left(S^1_{\emptyset,H(\boldsymbol{\beta})}, \Psi^1_{\emptyset,H(\boldsymbol{\beta})}(\cdot)\right), ..., \mathcal{P}\left(S^{\omega_{\emptyset,H(\boldsymbol{\beta})}}_{\emptyset,H(\boldsymbol{\beta})}, \Psi^{\omega_{\emptyset,H(\boldsymbol{\beta})}}_{\emptyset,H(\boldsymbol{\beta})}(\cdot)\right)\right] \\
\mathcal{S}_{\emptyset,H(\boldsymbol{\beta}')} &= \left[\mathcal{P}\left(S^1_{\emptyset,H(\boldsymbol{\beta}')}, \Psi^1_{\emptyset,H(\boldsymbol{\beta}')}(\cdot)\right), ..., \mathcal{P}\left(S^{\omega_{\emptyset,H(\boldsymbol{\beta}')}}_{\emptyset,H(\boldsymbol{\beta}')}, \Psi^{\omega_{\emptyset,H(\boldsymbol{\beta}')}}_{\emptyset,H(\boldsymbol{\beta}')}(\cdot)\right)\right],
\end{aligned}
$$

respectively. Similar to Step 3 in the proof of Proposition 6, it suffices to show that

$$
S^w_{\emptyset,H(\boldsymbol{\beta})} \neq S^{w'}_{\emptyset,H(\boldsymbol{\beta}')} \text{ for any } w \in \left\{1, ..., \omega_{\emptyset,H(\boldsymbol{\beta})}\right\} \text{ and } w' \in \left\{1, ..., \omega_{\emptyset,H(\boldsymbol{\beta}')}\right\}.
$$

Consider $\mathcal{P}\left(S^w_{\emptyset,H(\boldsymbol{\beta})}, \Psi^w_{\emptyset,H(\boldsymbol{\beta})}(\cdot)\right)$ in the sequence $\mathcal{S}_{\emptyset,H(\boldsymbol{\beta})}$. We have $S^w_{\emptyset,H(\boldsymbol{\beta})} \subseteq H(\boldsymbol{\beta})$. The definition of $H(\boldsymbol{\beta})$ in (6) yields $\boldsymbol{\beta}^T \mathbf{b}_i / a_i = 1$ for all $i \in S^w_{\emptyset,H(\boldsymbol{\beta})}$. As $\boldsymbol{\beta} = \left[\hat{\boldsymbol{\beta}}^T \, \mathbf{0}^T\right]^T$, the definition of $\hat{\mathbf{b}}_i$ in (4) implies $\hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i = 1$ for all $i \in S^w_{\emptyset,H(\boldsymbol{\beta})}$. Recall that the set $\{\hat{\mathbf{b}}_i : i \in S^w_{\emptyset,H(\boldsymbol{\beta})}\}$ contains $\hat{\kappa}$ linearly independent vectors. $\hat{\boldsymbol{\beta}}$ is the unique vector satisfying the linear equations $\hat{\boldsymbol{\beta}}^T \hat{\mathbf{b}}_i / a_i = 1$ for all $i \in S^w_{\emptyset,H(\boldsymbol{\beta})}$.

Similarly, we can show that $\hat{\boldsymbol{\beta}}'$ is the unique vector satisfying the linear equations $\hat{\boldsymbol{\beta}}'^T \hat{\mathbf{b}}_i / a_i = 1$ for all $i \in S^{w'}_{\emptyset,H(\boldsymbol{\beta}')}$. Therefore, $\hat{\boldsymbol{\beta}} \neq \hat{\boldsymbol{\beta}}'$ yields $S^w_{\emptyset,H(\boldsymbol{\beta})} \neq S^{w'}_{\emptyset,H(\boldsymbol{\beta}')}$. □

Appendix B. Branch-and-Bound Algorithm for $\mathcal{Q}$

Algorithm 3 adopts the branch-and-bound paradigm to find an optimal integral solution of $\mathcal{Q}$. At each node of the branch-and-bound tree, we need to solve an optimization problem in the form of

$$
\min \left\{ -\sum_{i \in S} a_i z_i + \Phi\left(\sum_{i \in S} \mathbf{b}_i z_i + \tilde{\mathbf{b}}\right) \;\middle|\; z_i \in [0,1] \; \forall i \in S \right\},
$$

where $S \subseteq I$ and $\tilde{\mathbf{b}} \in \mathbb{R}^\kappa_+$. Obviously, this problem has the same structure as the relaxation of $\mathcal{Q}$ and can be readily solved by Algorithm 1. Also note that Proposition 1 is utilized to fix some values of z_i in Line 13.

Algorithm 3: Solving $\mathcal{Q}$

Data: $a_i \in \mathbb{R}_+$ and $\mathbf{b}_i \in \mathbb{R}_+^\kappa$ for all $i \in I$, and a nondecreasing function $\Phi : \mathbb{R}_+^\kappa \mapsto \mathbb{R}$

Result: an optimal solution $(z_i^* \, \forall i \in I)$ and the optimal value v^* of $\mathcal{Q}$

1 let $z_i^* := 0$ for all $i \in I$, $v^* := \infty$, $m := 1$, $S_0^0 := \emptyset$, and $S_0^1 := \emptyset$;

2 **if** $|I \setminus S_m^0 \setminus S_m^1| > 10$ **then**

3 use Algorithm 1 to solve the following problem

$$\min \left\{ -\sum_{i \in I \setminus S_m^0 \setminus S_m^1} a_i z_i + \Phi \left(\sum_{i \in I \setminus S_m^0 \setminus S_m^1} \mathbf{b}_i z_i + \sum_{i \in S_m^1} \mathbf{b}_i \right) \;\middle|\; z_i \in [0,1] \; \forall i \in I \setminus S_m^0 \setminus S_m^1 \right\},$$

and let $(\bar{z}_i \, \forall i \in I \setminus S_m^0 \setminus S_m^1)$ and $\bar{v}$ denote its optimal solution and optimal value, respectively;

4 **else**

5 solve

$$\min \left\{ -\sum_{i \in I \setminus S_m^0 \setminus S_m^1} a_i z_i + \Phi \left(\sum_{i \in I \setminus S_m^0 \setminus S_m^1} \mathbf{b}_i z_i + \sum_{i \in S_m^1} \mathbf{b}_i \right) \;\middle|\; z_i \in \{0,1\} \; \forall i \in I \setminus S_m^0 \setminus S_m^1 \right\},$$

by enumerating all $\mathbf{z} \in \{0,1\}^{|I \setminus S_m^0 \setminus S_m^1|}$, and let $(\bar{z}_i \, \forall i \in I \setminus S_m^0 \setminus S_m^1)$ and $\bar{v}$ denote its optimal solution and optimal value, respectively;

6 **end**

7 **if** $\bar{v} - \sum_{i \in S_m^1} a_i \geq v^*$ **then** let $m := m - 1$ and go to Line 14;

8 **if** $z_i^* \in \{0,1\}$ *for all* $i \in I \setminus S_m^0 \setminus S_m^1$ **then**

9 let $v^* := \bar{v} - \sum_{i \in S_m^1} a_i$, $z_i^* := 0$ for all $i \in S_m^0$, $z_i^* := 1$ for all $i \in S_m^1$, and $z_i^* := \bar{z}_i$ for all $i \in I \setminus S_m^0 \setminus S_m^1$;

10 let $m := m - 1$ and go to Line 14;

11 **end**

12 let i^* be some $i \in I \setminus S_m^0 \setminus S_m^1$ such that $\bar{z}_{i^*} \in (0,1)$;

13 let $S_{m+1}^0 := S_m^0$, $S_{m+1}^1 := S_m^1 \cup \{i^*\} \cup \{i \in I : a_i > a_{i^*}, \mathbf{b}_i \leq \mathbf{b}_{i^*}\}$, $S_m^0 := S_m^0 \cup \{i^*\} \cup \{i \in I : a_i < a_{i^*}, \mathbf{b}_i \geq \mathbf{b}_{i^*}\}$, and $m := m + 1$;

14 **if** $m > 0$ **then** go to Line 2;

15 **return** $(z_i^* \, \forall i \in I)$ and v^*;

Appendix C. Solving the $\hat{\kappa}$-Variable Problems Decomposed from the Relaxation of $\mathcal{Q}$

According to the proofs of Proposition 6 and Theorem 3, solving the relaxation of $\mathcal{Q}$ boils down to solving at most $C_n^{\hat{\kappa}}$ problems in the form of $\mathcal{P}(S, \Psi(\cdot))$ defined in (8), where $|S| = \hat{\kappa}$ and $\hat{\mathbf{b}}_i$ for all $i \in S$ are linearly independent. In this section, we show how to solve these decomposed problems for the two applications implemented in Section 4.

C.1 Facility Location and Production in Section 4.1

For the κ-product facility location and production model, we have $\kappa = \hat{\kappa}$. Given the cubic production cost function, the decomposed problem with κ decision variables can be written as

$$\min_{\mathbf{z} \in [0,1]^\kappa} \left\{ -\sum_{i=1}^{\kappa} a_i z_i + \sum_{k=1}^{\kappa} \left(\sum_{i=1}^{\kappa} b_i^k z_i + B^k \right)^3 \right\}, \tag{21}$$

where $a_i > 0$, $b_i^k \geq 0$, and $B^k \in \mathbb{R}$ for any $i, k \in \{1, ..., \kappa\}$. Furthermore, $\mathbf{b}_i$ for all $i \in \{1, ..., \kappa\}$ are linearly independent. According to the KKT conditions, a solution $\mathbf{z}^* \in [0,1]^\kappa$ is optimal only if there exist $u_i, v_i \geq 0$ for all $i \in \{1, ..., \kappa\}$ such that

$$-a_i + \sum_{k=1}^{\kappa} 3b_i^k \left(\sum_{i=1}^{\kappa} b_i^k z_i^* + B^k \right)^2 + u_i - v_i = 0, \qquad u_i(z_i^* - 1) = 0, \qquad \text{and} \qquad v_i z_i^* = 0.$$

Therefore, an optimal solution to (21) must belong to the set

$$\bigcup_{\substack{S_0, S_1 \subseteq \{1, ..., \kappa\}, \\ S_0 \cap S_1 = \emptyset}} \mathcal{Z}(S_0, S_1),$$

where

$$\mathcal{Z}(S_0, S_1) = \left\{ \mathbf{z}^* \in [0,1]^\kappa \;\middle|\; \begin{array}{c} z_i^* = 0 \ \forall i \in S_0, \quad z_i^* = 1 \ \forall i \in S_1, \\ -a_i + \sum_{k=1}^{\kappa} 3b_i^k \left(\sum_{i=1}^{\kappa} b_i^k z_i^* + B^k \right)^2 = 0 \ \forall i \notin S_0 \cup S_1 \end{array} \right\}.$$

Next, we show how to obtain $\mathcal{Z}(S_0, S_1)$ for any $S_0, S_1 \subseteq \{1, ..., \kappa\}$ and $S_0 \cap S_1 = \emptyset$ through the following cases based on the value of $|S_0| + |S_1|$.

Case 1. Suppose that $|S_0| + |S_1| = \kappa$, i.e., $\{1, ..., \kappa\} \setminus S_0 \setminus S_1 = \emptyset$. It is straightforward that

$$\mathcal{Z}(S_0, S_1) = \{\mathbf{z}^* \in [0,1]^\kappa \mid z_i^* = 0 \ \forall i \in S_0, \ z_i^* = 1 \ \forall i \in S_1\}.$$

Case 2. Suppose that $|S_0| + |S_1| = \kappa - m \in \{1, ..., \kappa - 1\}$, i.e., $m \in \{1, ..., \kappa - 1\}$. WLOG, let $\{1, ..., \kappa\} \setminus S_0 \setminus S_1 = \{1, ..., m\}$. We have

$$\mathcal{Z}(S_0, S_1) = \left\{ \mathbf{z}^* \in [0, 1]^\kappa \;\middle|\; \begin{array}{c} z_i^* = 0 \ \forall i \in S_0, \quad z_i^* = 1 \ \forall i \in S_1, \\ -a_i + \sum_{k=1}^{\kappa} 3b_i^k \left(\sum_{i=1}^{m} b_i^k z_i^* + \sum_{j \in S_1} b_j^k + B^k \right)^2 = 0 \ \forall i \in \{1, ..., m\} \end{array} \right\}.$$

Obviously, $\{z_1^*, ..., z_m^*\}$ can be obtained by solving a system of m quadratic equations. If $m = 1$, z_1^* is a root of a quadratic function and has a closed-form solution. If $m \in \{2, ..., \kappa - 1\}$, the equation system can be easily solved by standard numerical methods. In our computational experiments, its numerical solutions are computed by calling the corresponding Mathematica function from C++ platform.

Case 3. Suppose that $|S_0| + |S_1| = 0$, i.e., $S_0 = S_1 = \emptyset$. Then

$$\mathcal{Z}(S_0, S_1) = \left\{ \mathbf{z}^* \in [0, 1]^\kappa \;\middle|\; -a_i + \sum_{k=1}^{\kappa} 3b_i^k \left(\sum_{i=1}^{\kappa} b_i^k z_i^* + B^k \right)^2 = 0 \ \forall i \in \{1, ..., \kappa\} \right\}.$$

Define

$$x_k = \sum_{i=1}^{\kappa} b_i^k z_i^* + B^k \quad \forall k \in \{1, ..., \kappa\},$$

i.e., $\mathbf{x} = \mathbf{M}\mathbf{z}^* + \mathbf{B}$, where $\mathbf{z}^* = [z_1^*, z_2^*, ..., z_\kappa^*]^T$, $\mathbf{x} = [x_1, x_2, ..., x_\kappa]^T$, $\mathbf{M} = [\mathbf{b}_1 \, \mathbf{b}_2 \cdots \mathbf{b}_\kappa]$, and $\mathbf{B} = [B^1, B^2, ..., B^\kappa]^T$. Recall that $\mathbf{b}_i$ for all $i \in \{1, ..., \kappa\}$ are linearly independent. Thus, $\mathbf{M}$ is invertible and $\mathbf{z}^* = \mathbf{M}^{-1}(\mathbf{x} - \mathbf{B})$. Then $\mathcal{Z}(S_0, S_1)$ is equivalent to

$$\mathcal{Z}(S_0, S_1) = \left\{ \mathbf{z}^* \in [0, 1]^\kappa \;\middle|\; \mathbf{z}^* = \mathbf{M}^{-1}(\mathbf{x} - \mathbf{B}), \ -a_i + \sum_{k=1}^{\kappa} 3b_i^k x_k^2 = 0 \ \forall i \in \{1, ..., \kappa\} \right\}.$$

Furthermore, let $y_k = x_k^2$ for all $k \in \{1, ..., \kappa\}$. The constraints $-a_i + \sum_{k=1}^{\kappa} 3b_i^k x_k^2 = 0$ for all $i \in \{1, ..., \kappa\}$ can be written as $3\mathbf{M}\mathbf{y} = \mathbf{a}$, where $\mathbf{y} = [y_1, y_2, ..., y_\kappa]^T$ and $\mathbf{a} = [a_1, a_2, ..., a_\kappa]^T$, which yields $\mathbf{y} = \mathbf{M}^{-1}\mathbf{a}/3$. As a result, $\mathcal{Z}(S_0, S_1)$ can be computed as

$$\mathcal{Z}(S_0, S_1) = \left\{ \mathbf{z}^* \in [0, 1]^\kappa \;\middle|\; \mathbf{z}^* = \mathbf{M}^{-1}(\mathbf{x} - \mathbf{B}), \ \mathbf{y} = \mathbf{M}^{-1}\mathbf{a}/3, \ x_k \in \{\sqrt{y_k}, -\sqrt{y_k}\} \ \forall i \in \{1, ..., \kappa\} \right\}.$$

C.2 Facility Location and Capacity Planning in Section 4.2

In this example, we have $\kappa = \hat{\kappa} = 2$. The decomposed problems can be written as follows:

$$\min_{z_1, z_2 \in [0,1]} \left\{ -\sum_{i=1}^{2} a_i z_i + \Phi \left(\sum_{i=1}^{2} b_i z_i, \sum_{i=1}^{2} c_i z_i \right) \right\}, \tag{22}$$

where $a_i > 0$ and $b_i, c_i \geq 0$ for $i = 1, 2$ and

$$\Phi(x, y) = \left(B + x + \sqrt{C + y}\right)^3$$

with $C \geq 0$.

For any $x^1, x^2, y^1, y^2 \geq 0$ and $\lambda \in [0, 1]$, let $x = \lambda x^1 + (1 - \lambda)x^2$ and $y = \lambda y^1 + (1 - \lambda)y^2$. As the square root function is concave, we have

$$\begin{aligned} B + x + \sqrt{C + y} &\geq \lambda \left(B + x^1 + \sqrt{C + y^1}\right) + (1 - \lambda) \left(B + x^2 + \sqrt{C + y^2}\right) \\ &\geq \min \left\{B + x^1 + \sqrt{C + y^1}, B + x^2 + \sqrt{C + y^2}\right\}. \end{aligned}$$

The monotonicity of the cubic function immediately yields

$$\begin{aligned} \Phi(x, y) = \left(B + x + \sqrt{C + y}\right)^3 &\geq \min \left\{ \left(B + x^1 + \sqrt{C + y^1}\right)^3, \left(B + x^2 + \sqrt{C + y^2}\right)^3 \right\} \\ = \min \left\{ \Phi(x^1, y^1), \Phi(x^2, y^2) \right\}, \end{aligned}$$

implying that $\Phi(x, y)$ is a quasi-concave function. Applying Proposition 2, the problem in (22) has an optimal solution with at most one fractional component. Therefore, it can be reformulated as

$$\min_{i \in \{1,2\}, \delta \in \{0,1\}} \{f(i, \delta)\},$$

where

$$f(i, \delta) = \min_{z_i \in [0,1]} \left\{ -a_i z_i - a_{3-i}\delta + \left(B + b_i z_i + b_{3-i}\delta + \sqrt{C + c_i z_i + c_{3-i}\delta}\right)^3 \right\}.$$

According to the KKT conditions, an optimal solution to this optimization problem can only be 0, 1, or some $z^* \in (0, 1)$ satisfying

$$-a_i + 3 \left(B + b_i z^* + b_{3-i}\delta + \sqrt{C + c_i z^* + c_{3-i}\delta}\right)^2 \left(b_i + \frac{c_i}{2\sqrt{C + c_i z^* + c_{3-i}\delta}}\right) = 0.$$

Obviously, the above equation is equivalent to a univariate polynomial equation and can be easily solved numerically. In our computational experiments, we obtain its numerical solutions by calling the corresponding Mathematica function from C++ platform.