A branch-and-price algorithm for the robust single-source capacitated facility location problem under demand uncertainty

Jaehyeon Ryu¹, Sungsoo Park¹,∗

Department of Industrial and Systems Engineering, KAIST, 291 Daehak-ro, Yuseong-gu, Daejeon 34141, Republic of Korea

Abstract

We consider the robust single-source capacitated facility location problem with uncertainty in customer demands. A cardinality-constrained uncertainty set is assumed for the robust problem. To solve it efficiently, we propose an allocation-based formulation derived by Dantzig-Wolfe decomposition and a branch-and-price algorithm. The computational experiments show that our branch-and-price algorithm outperforms CPLEX in many cases, which solves the ordinary robust reformulation. We also examine the trade-off relationship between the empirical probability of infeasibility and the additional costs incurred and observe that the robustness of solutions can be improved significantly with small additional costs.

Keywords: robust optimization, single source capacitated facility location problem, demand uncertainty, Dantzig-Wolfe decomposition, column generation, branch-and-price algorithm

1. Introduction

Facility location problem is one of the important combinatorial optimization problems arising in telecommunications, production-distribution systems, transportation, and many other industrial fields. The problem involves determining optimal locations of facilities and assignments of customers to the facilities with the least cost. A number of variations of the problem and their solution methods have been considered in the literature. Recently, facility location problems under parameter uncertainty have also been addressed and solved by many researchers.

We consider the single-source capacitated facility location problem (SSCFLP). In this problem, it is only allowed that each customer must be assigned to exactly one facility. Additionally, each facility has a capacity restriction so that it can serve a set of customers as long as the total demand of the assigned customers is within the capacity limit. The objective is to minimize the overall costs of opening the facilities and the assignment of customers to facilities.

The SSCFLP is strongly NP-hard (Cornuéjols et al., 1991; Gadegaard et al., 2018), which means there exist neither a pseudo-polynomial time algorithm nor a fully polynomial-time approximation scheme to solve it efficiently unless P=NP. Polynomial-time reduction from the uncapacitated facility location problem (Cornuéjols et al., 1991) or the node cover problem (Gadegaard et al., 2018) shows this negative theoretical result. However, many algorithms for the SSCFLP have been proposed, which usually fall into one of the categories of Lagrangian relaxation-based algorithms, heuristic algorithms, and branch-and-bound-based exact algorithms.

Lagrangian relaxation has been used, combined with branch-and-bound or heuristics, to obtain lower bounds on the optimal value. Klincewicz and Luss (1980) proposed Lagrangian relaxation whose relaxed problems are uncappedacitated facility location problems by dualizing the capacity constraints, and feasible

¹Corresponding author. Tel +82 42 350 3121

Email addresses: jhryu0340@gmail.com (Jaehyeon Ryu), sspark@kaist.ac.kr (Sungsoo Park)

¹Korea Advanced Institute of Science and Technology, Daejeon, the Republic of Korea

Preprint submitted to Elsevier
solutions were obtained by Lagrangian heuristics. Barceló and Casanovas (1984) presented a two-stage algorithm, whose first stage determines facility locations based on Lagrangian relaxation with the single-sourcing constraints relaxed and second stage solves generalized assignment problems. Pirkul (1987) and Sridharan (1993) also relaxed the single-sourcing constraints for their Lagrangian relaxation-based algorithm, whose subproblems are binary knapsack problems. Beasley (1993) proposed Lagrangian relaxation, dualizing both of the capacity constraints and single-sourcing constraints. Hindi and Pieńkosz (1999) applied greedy heuristics using Lagrangian relaxation and restricted neighborhood search to find solutions of the large-scale SSCFLP.

Heuristics have also been proposed to obtain high-quality feasible solutions of the SSCFLP in a short time. Rönqvist et al. (1999) identified feasible solutions from a repeated-matching algorithm based on three sets of closed facilities, unassigned customers, and pairs of each assigned customer and her facility, respectively. Delmaire et al. (1999) presented a hybrid heuristic algorithm with a greedy randomized adaptive search procedure (GRASP) and tabu search for the SSCFLP. Cortíñhal and Captivo (2003) incorporated tabu search into the procedures of Lagrangian heuristics. Ahuja et al. (2004) improved multi-exchange heuristics by exchanging the set of customers assigned to each facility. Furthermore, ant colony optimization by Chen and Ting (2008), scatter search by Contreras and Díaz (2008), and kernel search by Guastaroba and Speranza (2014) are proposed for the large-scale SSCFLP.

There have also been studies to solve the SSCFLP exactly by applying branch-and-bound-based algorithms. Neebe and Rao (1983) formulated the SSCFLP as a set partitioning problem and solved its linear programming relaxation (LP-relaxation) by a column generation approach at each node of the search tree. Holmberg et al. (1999) utilized Lagrangian relaxation, relaxing single-sourcing constraints, to obtain lower bounds in the branch-and-bound algorithm. Díaz and Fernández (2002) applied a branch-and-price algorithm with two levels of nodes; The first level nodes, children of the root node, are corresponding to the selection of opened facilities, and the second level nodes, children of the first level nodes, are corresponding to the allocation of customers, respectively. Recently, Yang et al. (2012) proposed a modified branch-and-cut algorithm with lifted cover inequalities and Fenchel cutting planes to solve the SSCFLP. Their branching scheme generates a pair of nodes at each level; one involves a small-sized sparse problem with some variables fixed to zero to get feasible solutions, and the other contains a dense problem to obtain lower bounds. Gadegaard et al. (2018) improved the algorithm of Yang et al. (2012) in terms of cut generations and local branching strategies.

Meanwhile, there has been much effort to handle facility location problems under uncertainty of parameters such as costs, demands, etc. It has been observed that an optimal solution to a deterministic problem can be inefficient or even infeasible to small changes of problem data (Ben-Tal and Nemirovski, 2000). Stochastic programming and robust optimization are two important approaches that have been used to handle parameter uncertainty.

Stochastic programming is based on the assumption that there are certain probability distributions of all or some of the parameters, which are known in advance. To introduce the overview of the models and solution algorithms for the stochastic facility location problems, we refer to Owen and Daskin (1998) and Savelsbergh (2004). One of the interesting cases of the stochastic facility location problems is demand uncertainty. It has been described using random variables, and therefore, capacity constraints now can be defined as chance-constraints with a probability level. Each chance-constraint states that the probability of the total demand of the customers assigned to a facility exceeding its capacity is less than a specified probability level. Laporte et al. (1994) introduced chance-constraints to the capacitated facility location problem with stochastic customer demands. They formulated it as a mixed-integer programming (MIP) problem and solved it by a branch-and-cut algorithm. Beraldi et al. (2004) assumed that the demands of emergency medical services follow the Poisson distribution, and they formulated the problem as a stochastic integer programming model with chance-constraints. Liu (2009) assumed that the distribution of customer demands of the SSCFLP can be Poisson or normal and defined the capacity restrictions as chance-constraints, which can be formulated as a mixed-integer nonlinear programming problem for the case of normally distributed demand uncertainty.

The additional costs to the objective function incurred by excessive demands at each facility are also considered for the SSCFLP. Albareda-Sambola et al. (2011) provided a formulation of the stochastic SSCFLP.
with a restriction on the number of assigned customers to each facility when each customer demand, restricted by whether it is necessary or not, follows the Bernoulli distribution. They added the expected value of additional costs, which can occur by reassigning customers to another facility, to the objective function. Then, it can be formulated as an MIP problem, and they solved the instances with at most 20 candidates of facilities and 60 customers by CPLEX. Bieniek (2015) extended the assumption on the distribution of demands to arbitrary discrete, continuous, or mixed distributions. The paper includes theoretical results for general distribution and computational experiments for a small instance with four facilities and twelve customers whose demands have exponential or Poisson distribution.

However, these stochastic programming approaches have some limitations. First, exact distributions of parameters are required for a stochastic programming formulation, but it is not easy to know the true distributions of the parameters practically. Moreover, even if the probability distributions can be assumed precisely, an optimal solution of the problem often cannot be obtained exactly and effectively by the existing methods. Such difficulty usually comes from non-linearity, sometimes non-convexity of the stochastic objective function, and the chance-constraints in the stochastic MIP problems.

Robust optimization can be an alternative approach for incorporating the uncertainty of parameters into optimization problems. An uncertainty set, instead of probabilistic information, is used to represent the range of parameter changes for robust optimization problems. For example, there are uncertainty sets such as simple interval uncertainty set (Sovest, 1973), ellipsoidal uncertainty set (Ben-Tal and Nemirovski, 1998, 2000), and cardinality-constrained uncertainty set (Bertsimas and Sim, 2003, 2004) have been considered.

There have been several results for robust facility location problems such as Snyder and Daskin (2006) and Gülpınar et al. (2013). Moreover, we refer to Baron et al. (2011) for a comprehensive review of the robust facility location problems. However, to the best of our knowledge, there has been little previous research for the robust optimization approach for the SSCFLP with demand uncertainty. Recently, Baron et al. (2019) proposed the almost robust optimization approach for it. This scenario-based, soft-constrained robust optimization framework allows a solution having a few infeasible scenarios. Their proposed decomposition algorithm could solve instances with at most 25 candidates of facilities and 50 customers in about five minutes.

Like the deterministic SSCFLP, its robust counterpart also can be reformulated using the Dantzig-Wolfe decomposition, as we suggest in this paper. Because the resulting reformulation has exponentially many variables, it cannot be solved directly. Column generation and branch-and-price method can be used to solve such a problem. They have been used successfully to solve many difficult combinatorial optimization problems with many variables. We refer to Barnhart et al. (1998), Desrosiers and Lübbecke (2010), and Gamrath (2010) for further details of the branch-and-price algorithm.

There have been many successful trials to solve large-scale MIP problems using the branch-and-price algorithm. Savelsbergh (1997) solved the generalized assignment problem using a branch-and-price algorithm. Díaz and Fernández (2002) applied a branch-and-price algorithm to solve the deterministic SSCFLP. Ceselli and Righini (2003) used a branch-and-price algorithm to solve the capacitated p-median problem, which is one of the location problems having the same capacity restrictions and single-source restrictions like the SSCFLP. Klose and Görtz (2007) solved the capacitated facility location problem without the single-source constraints using a branch-and-price algorithm. Lee et al. (2012) proposed a branch-and-price algorithm for the robust network design problem without flow bifurcations using the cardinality-constrained uncertainty set for demands. Moreover, Lee et al. (2012) also presented a branch-and-price-and-cut algorithm for the robust vehicle routing problem with travel time and demand uncertainty.

In this paper, we consider the SSCFLP with demand uncertainty using a robust optimization perspective. We assume that the demand of each customer belongs to a specified interval uncertainty set. The cardinality-constrained uncertainty set (Bertsimas and Sim, 2003, 2004) is used to describe the demand uncertainty of the robust SSCFLP. This uncertainty set is less conservative than the simple interval uncertainty set (Sovest, 1973), and linearity of the formulation can be preserved, unlike the ellipsoidal uncertainty set (Ben-Tal and Nemirovski, 1998, 2000).

After reformulating the problem using the Dantzig-Wolfe decomposition, we propose a branch-and-price algorithm to solve the robust SSCFLP. We will show how the uncertainty of demands can be isolated into the subproblem in the column generation procedure. Therefore, overall optimization is not affected by
the uncertainty of demands. We also consider branching schemes, variable fixing, and early termination to improve the performance of the algorithm. Computational experiments show that the algorithm can solve the robust SSCFLP fast compared to the traditional reformulation approach. Moreover, we make observations by simulation that the robustness of the solutions is improved by incorporating demand uncertainty.

The rest of the paper is organized as follows. In section 2, we consider the traditional reformulation of the robust SSCFLP and the Dantzig-Wolfe decomposition-based reformulation. Section 3 explains the technical details of the branch-and-price algorithm to solve the reformulation. Section 4 presents the computational results of our branch-and-price algorithm compared to the traditional MI P reformulation. Section 5 presents the details of the branch-and-price algorithm to solve the reformulation. Section 4 gives computational results that summarizes the result of our research.

2. Formulations of the robust SSCFLP

In this section, we introduce MIP formulations of the robust SSCFLP with a cardinality-constrained uncertainty set for demands. We also present an allocation-based formulation that can be obtained using the Dantzig-Wolfe decomposition.

2.1. Robust SSCFLP with cardinality-constrained demand uncertainty

We first introduce notation as follows. Let \( M = \{1, \ldots, m\} \) be a set of candidate facility locations and \( N = \{1, \ldots, n\} \) be a set of customers. Let \( f_i \) be the set-up cost of opening facility and \( s_i \) be the capacity of the facility at location \( i \in M \). Let \( d_{ij} \) be the demand of customer \( j \in N \) and \( c_{ij} \) be the allocation cost of assigning customer \( j \in N \) to facility \( i \in M \). Without loss of generality, we assume that these parameters are nonnegative integers.

Then, we can formulate the SSCFLP as follows:

\[
\begin{align*}
\text{(P)} \quad & \text{minimize} & & \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} + \sum_{i \in M} f_i y_i \\
\text{subject to} & & & \sum_{j \in N} d_{ij} x_{ij} \leq s_i y_i, \forall i \in M, \\
& & & \sum_{i \in M} x_{ij} = 1, \forall j \in N, \\
& & & x_{ij} \leq y_i, \forall i \in M, j \in N, \\
& & & x_{ij} \in \{0, 1\}, \forall i \in M, j \in N, \\
& & & y_i \in \{0, 1\}, \forall i \in M, \end{align*}
\]

where the binary variable \( x_{ij} \) is equal to one if customer \( j \in N \) is served by facility \( i \in M \) and zero otherwise, and the binary variable \( y_i \) is equal to one if facility \( i \in M \) is opened, and zero otherwise. The objective function (1) minimizes the sum of total fixed costs and total assignment costs. Constraints (2) ensure that the total demand of the customers assigned to a facility should not exceed the capacity of the facility. Constraints (3) ensure that each customer must be assigned to exactly one facility. Constraints (4) are redundant, but the lower bound obtained by the LP-relaxation of (P) can be strengthened by adding these constraints.

Now, we formulate the robust SSCFLP under demand uncertainty. A cardinality-constrained uncertainty set \( \{d \in \mathbb{R}_+^n | d_j = d_j^0 + b_j, \sum_{j \in N} |v_j| \leq \Gamma, |v_j| \leq 1\} \) for facility \( i \in M \) is defined as \( U_i := \{d \in \mathbb{R}_+^n | d_j = d_j^0 + b_j v_j, \sum_{j \in N} |v_j| \leq \Gamma, |v_j| \leq 1\} \) for facility \( i \in M \).
Here, the capacity constraints (2) of (P) can be expressed as follows so that the demand uncertainty is reflected using the cardinality-constrained uncertainty set $U_i^d$.

$$\sum_{j \in N} d_j x_{ij} \leq s_i y_i, \quad \forall d \in U_i^d, i \in M.$$  

(7)

These constraints are equivalent to the following nonlinear constraints.

$$\sum_{j \in N} d_j x_{ij} + \max_{R \subseteq N, |R| \leq \Gamma_i} \sum_{j \in R} b_j x_{ij} \leq s_i y_i, \quad \forall i \in M.$$  

(8)

We note that these constraints are also equivalent to the following:

$$\sum_{j \in N} d_j x_{ij} + \sum_{j \in R} b_j x_{ij} \leq s_i y_i, \quad \forall R \subseteq N, |R| \leq \Gamma_i, i \in M.$$  

(9)

Then, the problem can be formulated as the following MIP problem by replacing the capacity constraints with constraints (9).

$$\text{(RP1)} \quad \text{Minimize} \quad \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} + \sum_{i \in M} f_i y_i$$

subject to (3) − (6), (9).

Moreover, we can obtain an alternative MIP formulation of (RP1). Bertsimas and Sim (2003, 2004) showed that constraints (8) can be reformulated using strong duality to the inner maximization term as follows:

$$\sum_{j \in N} d_j x_{ij} + \sum_{j \in R} b_j x_{ij} \leq s_i y_i, \quad \forall R \subseteq N, |R| \leq \Gamma_i, i \in M.$$  

(10)

$$q_i + p_{ij} \geq b_i x_{ij}, \quad \forall i \in M, j \in N, \quad p_{ij} \geq 0, \quad \forall i \in M, j \in N,$$  

(11) (12)

$$q_i \geq 0, \quad \forall i \in M.$$  

(13)

Therefore, the robust SSCFLP can be reformulated as the following MIP problem:

$$\text{(RP2)} \quad \text{minimize} \quad \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} + \sum_{i \in M} f_i y_i$$

subject to (3) − (6), (10) − (13).

(RP2) is a mixed-integer programming problem with $mn + m$ additional variables and $2mn + 2m$ additional constraints than problem (P), which makes it more difficult to solve than the deterministic problem (P). We also note that the lower bounds obtained by solving the LP-relaxation of (RP1) and LP-relaxation of (RP2) are the same. Instead of solving (RP2), we propose an allocation-based formulation that isolates the difficulty arising from the demand uncertainty into the subproblem. The allocation based formulation provides a stronger LP relaxation bound than the LP relaxation of (RP2).

2.2 Allocation-based formulation of the robust SSCFLP

We derive an allocation-based formulation of the robust SSCFLP by employing the Dantzig-Wolfe decomposition. We take (3) as the joint constraints, and the other constraints of (P) with the robust capacity constraints (5) substituting the ordinary capacity constraints (2) can be used for decomposition. Let $\Omega_i := \{(x_1, \ldots, x_n, y_i) \in \{0, 1\}^{n+1} | \sum_{j \in N} d_j x_{ij} + \max_{R \subseteq N, |R| \leq \Gamma_i} \sum_{j \in R} b_j x_{ij} \leq s_i y_i \}$ for $i \in M$. Then, $\Omega_i$
consists of \((x_i, 0) = (0, 0)\) and \((x_i, 1)\) for \(R \in S'\), where \(x_i\) is the incidence vector of set \(R \subseteq N\) of customers and \(S'\) is a set of all possible allocations of customers to facility \(i \in M\), i.e.

\[
S' := \{ R \subseteq N | \sum_{j \in R} d_j + \max_{S \subseteq \Gamma, |S| \leq \Gamma} \sum_{j \in S} b_j \leq s_i, \quad \forall i \in M. \tag{14} \]

We can define associated binary variables \(v_i\) for \((0, 0)\) and binary variables \(z_{ik}\) for \((x_i, 1)\) for \(R \in S', i \in M\), respectively, with constraints \(\sum_{R \in S'} z_{ik} + v_i = 1\) for \(i \in M\). Then, the binary variables \(x_{ij}\) and \(y_i\) can be expressed as \(x_{ij} = \sum_{R \in S'} z_{ik}^j R = \sum_{R \in S'} z_{ik}^j R\) and \(y_i = 0 \cdot v_i + \sum_{R \in S'} 1 \cdot z_{ik}^1 R = \sum_{R \in S'} z_{ik}^1 R\), respectively.

Now, substituting for \(x_{ij}\) and \(y_i\), variables in constraints \(3\) and the objective function \(1\) leads to the reformulation of the robust SSCFLP as follows:

\[
\text{(AP) Minimize } \sum_{i \in M} \sum_{R \in S'} c^i_R z^j_R \tag{15} \\
\text{subject to } \sum_{i \in M} \sum_{R \in S'} z^j_R = 1, \quad \forall j \in N, \tag{16} \\
\sum_{R \in S'} z^j_R + v_i = 1, \quad \forall i \in M, \tag{17} \\
v_i \in \{0, 1\}, \quad \forall i \in M, \tag{18} \\
z^j_R \in \{0, 1\}, \quad \forall R \in S', i \in M, \tag{19}
\]

where \(c^j_R := \sum_{j \in R} c_{ij} + f_i\) for \(R \in S', i \in M\). The binary variable \(v_i\) is equal to one if facility \(i \in M\) is not opened, and zero otherwise. Also, The binary variable \(z_{ik}^j R\) is equal to one if opened facility \(i \in M\) covers customers in \(R\), and zero otherwise for \(R \in S', i \in M\). The objective function \(1\) minimizes the total fixed costs and assignment costs of all opened facilities. Constraints \(14\) ensure that all customers must be covered by exactly one combination of customers at each facility. Constraints \(17\) ensure that each facility must be closed or opened, and it must take exactly one combination of customers when it is opened.

3. Branch-and-price algorithm

In this section, we present a branch-and-price algorithm for the allocation-based formulation \(\text{(AP)}\) of the robust SSCFLP.

3.1. Linear programming master problem

The LP-relaxation of the allocation-based formulation \(\text{(AP)}\) can be obtained by dropping the integrality restrictions on the variables as follows:

\[
\text{Minimize } \sum_{i \in M} \sum_{R \in S'} c^i_R z^j_R \tag{20} \\
\text{subject to } \sum_{i \in M} \sum_{R \in S'} z^j_R = 1, \quad \forall j \in N, \tag{21} \\
\sum_{R \in S'} z^j_R + v_i = 1, \quad \forall i \in M, \tag{22} \\
0 \leq v_i \leq 1, \quad \forall i \in M, \tag{23} \\
0 \leq z^j_R \leq 1, \quad \forall R \in S', i \in M, \tag{24}
\]

We can compare the strength of the LP-relaxation of \(\text{(AP)}\) and LP-relaxation of \(\text{(RP1)}\) and its reformulation \(\text{(RP2)}\).

**Proposition 1.** The LP-relaxation of \(\text{(AP)}\) has the same optimal value as that of the LP-relaxation of \(\text{(RP1)}\) augmented with all valid inequalities describing the convex hull of \(\Omega_i, i \in M\).
Proof. The LP-relaxation of (AP) can be obtained by substituting $x_{ij} = \sum_{R \in S^i} y_i z_{ij}^R$, $y_i = \sum_{R \in S^i} z_{ij}^R$, and constraints (21)–(23). This is equivalent to substituting $(x_{i1}, \cdots, x_{in}, y_i) \in \text{conv}(\Omega_i)$.

Because the lower bounds obtained by solving the LP-relaxation of (RP1) and LP-relaxation of (RP2) are the same, we can see that the LP-relaxation of (AP) provides a stronger lower bound than that of the LP-relaxation of (RP2). We also note that the LP-relaxation bound of (AP) is the same as the Lagrangian dual bound when the joint constraints (3) are dualized.

The LP-relaxation of (AP) can be modified further for improving computational efficiency. Constraints (20) can be replaced by inequalities:

$$\sum_{i \in M} \sum_{R \in S^i} x_{ij} \geq 1, \quad \forall j \in N,$$

(24)

Because all set-up costs and allocation costs are nonnegative, there exists an optimal solution that also satisfies constraints (20). Constraints (24) restrict the corresponding dual variables to be nonnegative, which can make the column generation procedure more stable compared to using unrestricted dual variables.

Constraints (21) can be replaced by inequalities:

$$- \sum_{R \in S^i} z_{ij}^R \geq -1, \quad \forall i \in M,$$

(25)

because the variables $v_i$ can be regarded as slack variables in constraints (21). In addition to this, constraints (23) can be replaced by inequalities:

$$z_{ij}^R \geq 0, \quad \forall R \in S^i, i \in M,$$

(26)

because of constraints (25).

As a result, the LP-relaxation of (AP) can be stated as the following linear programming master problem:

$$\text{(MP)} \quad \text{Minimize } \sum_{i \in M} \sum_{R \in S^i} c_{ij}^R z_{ij}^R$$

subject to

$$\sum_{i \in M} \sum_{R \in S^i} x_{ij} \geq 1, \quad \forall j \in N,$$

(27)

$$- \sum_{R \in S^i} z_{ij}^R \geq -1, \quad \forall i \in M.$$

(28)

$$z_{ij}^R \geq 0, \quad \forall R \in S^i, i \in M.$$

(29)

3.2. Restricted master problem and subproblem

We cannot solve (MP) directly since it has exponentially many variables. We suppose that we have a subset $R^i$ of $S^i$ for $i \in M$ which provides a feasible solution to (MP). Then, the following restricted problem (RMP) can be obtained:

$$\text{(RMP)} \quad \text{Minimize } \sum_{i \in M} \sum_{R \in R^i} c_{ij}^R z_{ij}^R$$

subject to

$$\sum_{i \in M} \sum_{R \in R^i} x_{ij} \geq 1, \quad \forall j \in N,$$

(27)

$$- \sum_{R \in R^i} z_{ij}^R \geq -1, \quad \forall i \in M.$$

(28)

$$z_{ij}^R \geq 0, \quad \forall R \in R^i, i \in M.$$

(29)

We solve (RMP) by the simplex method and obtain an optimal solution $z^*$ with optimal value $Z^*$. Let $\lambda \in \mathbb{R}^n_+$ and $\mu \in \mathbb{R}^m_+$ be a dual optimal solution corresponding to constraints (27) and (28), respectively.

During column generation, a column with a negative reduced cost is generated and added to (RMP) iteratively. This procedure continues until an optimal solution of (RMP) becomes also optimal for (MP).

7
The reduced cost of a variable $z_i^R$ is $\text{RC}_i,R(\lambda, \mu) := \sum_{j \in R} (c_{ij} - \lambda_j) + f_i + \mu_i \text{ for each } R \in S^i, i \in M$. We then try to find a column having a negative reduced cost by solving the following subproblem:

**Sub-1**  
Maximize $\xi^i := \sum_{j \in N} (\lambda_j - c_{ij})x_{ij}$  
subject to $d_jx_{ij} + \max_{R \subseteq S, |R| \leq \Gamma} \sum_{j \in R} b_jx_{ij} \leq s_i$,  
$x_{ij} \in \{0, 1\}, \forall j \in N$.

This problem is the robust binary knapsack problem with cardinality-constrained weight uncertainty. If every

$$\min_{R \in S^i} \text{RC}_i,R(\lambda, \mu) = -\xi^i + f_i + \mu_i$$

has a nonnegative value for $i \in M$, an optimal solution of (RMP) is also an optimal solution of (MP). Otherwise, if $\text{RC}_i,R(\lambda, \mu) < 0$, an optimal solution of (Sub-1) generates a column which has the smallest negative reduced cost among columns involving variable $i$ for $i \in M$.

Bertsimas and Sim (2003) showed that the robust BKP can be solved by solving the ordinary BKP$s at most $n + 1$ times. Lee et al. (2012) reduced the number of iterations to at most $n - \Gamma_i + 1$ times, and we apply it to solve (Sub-1). Let $C^i$ be a feasible solution set of (Sub-1) for $i \in M$. We assume that the values of the maximum possible deviation from the nominal demand are listed in nonincreasing order, and define a dummy value $b_{n+1} = 0$, i.e. $b_1 \geq b_2 \geq \cdots \geq b_n \geq b_{n+1} = 0$. We define a set $N^+ = N \cup \{n + 1\}$ and sets $N_i = \{1, \ldots \}$ for all $l \in N^+$. We then define $C^i = \{x_i \in \{0, 1\}^n | \sum_{j \in N} d_jx_{ij} + \sum_{j \in N \cap N^+} (b_j - b_i)x_{ij} \leq s_i - \Gamma_i b_l\}$ for $l \in \{\Gamma_i, \Gamma_i + 1, \ldots, n - 1, n + 1\}$. Then, $C^i$ can be obtained using the solution sets of ordinary binary knapsack problems (BKP).

**Proposition 2.** $C^i = \cup_{l \in \{\Gamma_i, \Gamma_i + 1, \ldots, n - 1, n + 1\}} C^i_l$

**Proof.** We refer to Lee et al. (2012) for the proof. \qed

Proposition 2 implies that we can solve (Sub-1) by solving BKP$s $n - \Gamma_i + 1$ times and taking the best solution among the optimal solutions to BKP$s.

The BKP can be solved by a branch-and-bound algorithm or a dynamic programming approach. Pisinger (1997) has provided the minknap algorithm based on the dynamic programming with pseudo-polynomial time complexity of $O(ns_i)$. Moreover, Martello et al. (1999) showed that the minknap algorithm solved the BKP faster than the other algorithms based on a branch-and-bound algorithm only. As mentioned previously, we solve the RBKP by solving the BKP $n - \Gamma + 1$ times, and it has pseudo-polynomial time complexity of $O((n - \Gamma + 1)ns_2)$.

We note that we may solve the MIP reformulation of the RBKP using Bertsimas and Sim’s approach (Bertsimas and Sim, 2003, 2004). However, Monaci et al. (2013) reported that the algorithm of Lee et al. (2012) solved the RBKP more effectively than CPLEX, which solved the MIP reformulation of the RBKP.

### 3.3. Branching scheme

If the optimal solution to (MP) has fractional values, we need to branch. However, direct branching on $z^R_i$ variables is not desirable. For example, if we branch on a variable $z_R^i$, two nodes are generated; one has $z_R^i = 0$, and the other has $z_R^i = 1$. If $z_R^i$ is fixed to zero, we need to make sure that the column for $z_R^i$ will not be generated again in subsequent column generation procedure, which is a nontrivial task. Such branching scheme also divides the feasible solution set unevenly. Díaz and Fernández (2002) discuss this defect of branching on the $z_R^i$ variables directly.

Instead, we use branching on the variables of (RP1) directly as suggested in Ceselli and Ricciatto (2005). Let $z^*$ be an optimal solution of (MP). Then, the value of $x$ and $y$ variables can be obtained as $x^*_{ij} = \sum_{R \subseteq S, j \in R} z^*_R$ for $i \in M, j \in N$, and $y^*_i = \sum_{R \subseteq S^i} z^*_R$ for $i \in M$ as shown in section 2.2. We note that $x$ and $y$ variables are integral if and only if $z$ variables are integral.
Ceselli and Righini (2005) used branching on $x$ variables only for the capacitated p-median problem, which is a variation of facility location problem. However, our preliminary testing showed that branching on $y$ variables first and then on $x$ variables gives better results. Therefore, we do branching on $x$ variables when all $y$ variables have integer values. Holmberg et al. (1999) also discussed some advantages and disadvantages of each branching scheme for the SSCFLP.

When we branch on $y$ variables, we branch on the variable $y_i$ having value closest to 0.5 among the candidate $y$ variables for branching. We set $y_i = 0$ on one branch, and $y_i = 1$ on the other branch.

If all $y$ variables are integer-valued and there are some fractional $x$ variables, we branch on $x$ variables. Let $x^*$ be the current fractional solution. We identify a customer $j'$ and use generalized upper bound (GUB) dichotomy (Savelesbergh, 1997) on the variables $x_{ij'}$ for all $i \in M$.

Let $N^* \subseteq N$ be the set of customers $j$ such that $x_{ij} > 0$ for more than one $i \in M$. For each $j \in N^*$, we divide $M$ into four sets $M_j^{11}$, $M_j^{12}$, $M_j^{21}$, and $M_j^{22}$ as follows. First, we divide $M$ into disjoint sets $M_j^1$ and $M_j^2$ for each $j \in N$, such that $i \in M_j^1$ if $x_{ij} > 0$, and $i \in M_j^2$ otherwise. Second, $M_j^1$ is divided into disjoint sets $M_j^{11}$ and $M_j^{12}$ such that $M_j^{11}$ minimizes $\sum_{i \in M_j^{11}} x_{ij}^* - 0.5$. We solve the following ordinary knapsack problem:

$$\kappa_j := \max_{M_j^{11} \subseteq M_j^1} \left\{ \sum_{i \in M_j^{11}} x^*_{ij} \mid \sum_{i \in M_j^{11}} x_{ij}^* \leq 0.5 \right\}, \quad j \in N^*,$$

to divide set $M_j^1$ for each $j \in N^*$. Third, choose customer $j' = \arg \min_{j \in N^*} |\kappa_j - 0.5|$ to make $\sum_{i \in M_j^{11}} x_{ij'}$ close to 0.5. Finally, divide $M_j^2$ into disjoint sets $M_j^{21}$ and $M_j^{22}$ to be of the same size. We then branch on $x_{ij'}$ for all $i \in M$. We set $x_{ij'} = 0$ for $i \in M_j^{11} \cup M_j^{21}$ on one branch, and we set $x_{ij'} = 0$ for $i \in M_j^{12} \cup M_j^{22}$ on the other.

We need to reflect the effect of some fixed variables to (RMP) and modify subproblem during subsequent column generation procedure. If variable $y_i$ is fixed to zero i.e. $y_i = 0$, then we set the upper bounds of $z^R_{ji}$ variables to zero for all $R \in R^i$, and we do not solve (Sub-i) during the column generation. Meanwhile, if variable $y_i$ is fixed to one, i.e. $y_i = 1$, inequality of constraints (28) of (RMP) is replaced by equality. As a result, the dual variable $\mu_i$ becomes free without nonnegativity, but other constraints of (RMP) are not changed and (Sub-i) still remains to be the robust BKP.

If variable $x_{ij}$ is fixed to zero, i.e. $x_{ij} = 0$, then we fix the upper bounds of $z_{ji}^R$ variables to zero for all $R \in R^i$ satisfying $j \in R$. Also, the subproblem (Sub-i) does not generate the column with $x_{ij} = 1$ during the column generation by setting the objective coefficient of $x_{ij}$ to some negative value. We note that we do not need to consider the case of fixing variable $x_{ij}$ to one because GUB dichotomy has been adopted for branching on $x$ variables.

### 3.4. Early termination and variable fixing

An optimal dual solution of (RMP) can be used to facilitate the branch-and-price procedure. In this section, we consider how the column generation can be terminated earlier before (MP) is completely optimized and how to fix the values of some variables.

The dual problem of (MP) with the dual variables $\lambda \in \mathbb{R}_+^n$ and $\mu \in \mathbb{R}_+^m$ is as follows:

\begin{align}
\text{(DMP)} \quad \text{Maximize} & \quad \sum_{j \in N} \lambda_j - \sum_{i \in M} \mu_i \\
\text{subject to} & \quad \sum_{j \in R} \lambda_j - \mu_i \leq c^R_i, \quad \forall R \in S^i, i \in M, \\
& \quad \lambda_j \geq 0, \quad \forall j \in N, \\
& \quad \mu_i \geq 0, \quad \forall i \in M,
\end{align}

where $c^R_i = f_i + \sum_{j \in R} c_{ij}$ for $R \in S^i, i \in M$. The dual of the restricted master problem (RMP) can be obtained by substituting $S^i$ by $R^i$ in constraints (30). We call it (DRMP).
Let \((\lambda^*, \mu^*)\) be an optimal solution of (DRMP). We note that the minimum reduced cost of variables \(z_R^i, R \in S^i\) for each \(i \in M\) is equal to \(\mu^*_i + f_i - \xi^*\), where \(\xi^*\) is optimal value of (Sub-i) with \(\lambda = \lambda^*\). Let \(\nu^* \in \mathbb{R}^m\) be a vector where \(\nu^*_v := \mu^*_v + \min\{f_i - \xi^*, 0\}\). Then, \((\lambda, \mu) = (\lambda^*, \mu^* - \nu^*)\) is a feasible solution to (DMP).

**Proposition 3.** \((\lambda, \mu) = (\lambda^*, \mu^* - \nu^*)\) is a feasible solution to (DMP).

*Proof.* For each \(i \in M\), \(\mu^*_i - \nu^*_i = -\min\{f_i - \xi^*, 0\} \geq 0\). Also, constraints (30) are equivalent to \(-\mu_i \leq \min_{R \in S^i}(c_R^i - \sum_{j \in R} \lambda_j) = f_i + \min_{R \in S^i}(c_R^i - \lambda_j - f_i) = f_i - \xi^i\) for \(i \in M\). We can see that the nonnegative vector \((\lambda, \mu)\) with \(\lambda = \lambda^*\) and \(\mu_i = -\min\{f_i - \xi^*, 0\}, i \in M\) satisfies these constraints obviously. Therefore, \((\lambda, \mu) = (\lambda^*, \mu^* - \nu^*)\) is a feasible solution to (DMP).

As a result, the objective value \(\sum_{j \in N} \lambda^*_j - \sum_{i \in M} (\mu^*_i - \nu^*_i) = Z + \sum_{i \in M} \nu^*_i\) can provide a lower bound on the optimal value of (MP), where \(Z\) is the optimal value to (RMP). If \(Z + \sum_{i \in M} \nu^*_i\) is greater than the current incumbent value, the column generation is terminated and the node is pruned.

Moreover, we can fix the value of some \(x\) and \(y\) variables to reduce the solution space although only \(z\) variables appear in (MP) and (RMP). The reduced costs of \(x\) and \(y\) variables can be computed by adding the constraints \(-x_{ij} + \sum_{R \in S^i} y^R_{ik} = 0, x_{ij} \geq 0\) for \(i \in M, j \in N\) and \(-y_i + \sum_{R \in S^i} z^R_i = 0, y_i \geq 0\) for \(i \in M\) to (MP), respectively, when they have zero values. Also, the reduced cost of surplus \(v\) variables, where \(v_i = 1 - \sum_{R \in S^i} z^R_i = 1 - y_i, i \in M\), also can be computed by adding the constraints \(-v_i + \sum_{R \in S^i} z^R_i = -1, v_i \geq 0\), for \(i \in M\) to (MP). We note that fixing \(v_i\) to zero is equivalent to fixing \(y_i\) to one. Let \(\delta \in \mathbb{R}^{m \times n}, \rho \in \mathbb{R}^m,\) and \(\tau \in \mathbb{R}^m\) denote an optimal dual solution corresponding to the coupling constraints for \(x, y, \) and \(v\) variables, respectively. Because only these coupling constraints have \(x, y, \) and \(v\) variables, the reduced cost of \(x_{ij}\) is equal to \(0 - (-1)\delta_{ij} = \delta_{ij}\) for \(i \in M, j \in N\), the reduced cost of \(y_i\) is equal to \(0 - (-1)\rho_i = \rho_i\), and the reduced cost of \(v_i\) is equal to \(0 - (-1)\tau_i = \tau_i\).

This technique is based on the approach of De Aragão and Uchoa (2003) and Fukasawa et al. (2006). However, explicitly adding these constraints in the problems not only increases the size of the problem but changes the structure of the master problem and the subproblem. Moreover, these explicit coupling constraints can amplify degeneracy problem of dual feasible solutions, which can hamper the convergence of the column generation procedure (Lee et al. 2012). Therefore, we apply an alternative approach without using the explicit coupling constraints.

When \(x_{ij} = 0\) for some \(i \in M, j \in N\) in an optimal solution to (MP), we assume that the coupling constraint \(-x_{ij} + \sum_{R \in S^i} z^R_i = 0\) is included in (MP). Let \(S^j\) be a set of all possible customer allocations including customer \(j\) to facility \(i\) i.e. \(S^j := \{R \in S^i \mid j \in R\}\). Let \(\xi^j\) be the optimal value of (Sub-i) when \(x_i\) is fixed to one. It can be solved like the original subproblems because we can solve the ordinary knapsack problem \(n - \Gamma_1 + 1\) times after fixing \(x_j = 1\) when we solve (Sub-i). The following proposition shows that the reduced cost of \(x_{ij}\) can be calculated without adding the additional explicit coupling constraints to (MP).

**Proposition 4.** Let \((\lambda^*, \mu^*)\) be an optimal solution to (DMP). For given \(i \in M\) and \(j \in N\), let \(\delta^j_{ij} = -\xi^j + f_i + \mu^*_i\). Then, \((\lambda^*, \mu^*, \delta^j_{ij})\) is an optimal solution to the dual problem of (MP) with the coupling constraints \(-x_{ij} + \sum_{R \in S^j \setminus R} z^R_i = 0, x_{ij} \geq 0\). Moreover, the reduced cost of \(x_{ij}\) is equal to \(\delta^j_{ij}\).

*Proof.* After augmenting the coupling constraint to (MP), constraints (30) for \(i \in M\) are replaced by \(\delta_{ij} + \sum_{k \in R} \lambda_k - \mu_i \leq c^R_{ij} \) for \(R \in S^j\), and \(\sum_{k \in R} \lambda_k - \mu_i \leq c^R_{ij} \) for \(R \in S^i \setminus S^j\). The first constraints are equivalent to \(\delta_{ij} \leq \min_{R \in S^j}(\sum_{k \in R} \lambda_k - \mu_i + c^R_{ij})\). Its right-hand side is equal to \(-\max_{R \in S^j}(\sum_{k \in R} \lambda_k - \mu_i + c^R_{ij})\). Also, the dual constraint corresponding to \(x_{ij}\) is \(-\delta_{ij} \leq 0\), and \(\delta^j_{ij} \geq -\xi^j + f_i + \mu^*_i\) is feasible to this constraint. Hence, \((\lambda^*, \mu^*, \delta^j_{ij})\) is a dual feasible solution. Because the dual objective function is independent of \(\delta_{ij}\), \((\lambda^*, \mu^*, \delta^j_{ij})\) is an optimal solution to the dual problem of (MP) with the coupling constraints. Thus, \(\delta^j_{ij}\) is the reduced cost of \(x_{ij}\).

Let \(Z\) be the optimal value of (MP) and \(\overline{Z}\) be the value of the current incumbent solution to (AP). If \(x_{ij}\) is equal to zero with reduced cost \(\delta^j_{ij}\), and \(\overline{Z} + \delta^j_{ij}\) is greater than \(\overline{Z}\), then there exists an optimal solution to (AP) with \(x_{ij} = 0\). Therefore, we can fix the value of \(x_{ij}\) to zero in subsequent branch-and-price procedure.
Variable fixing for $y_i = 0$, and $y_i = 1$ can be done similarly. However, we present the next two propositions for completeness. When $y_i = 0$ for some $i \in M$ in an optimal solution to (MP), we assume that the coupling constraints $-y_i + \sum_{R \in S_i} z_R = 0$, $y_i \geq 0$ are included in (MP). The following proposition shows that the reduced cost of $y_i$ can be calculated without adding the additional explicit coupling constraints to (MP).

**Proposition 5.** Let $(\lambda^*, \mu^*)$ be an optimal solution to (DMP). For given $i \in M$, let $\rho^*_i = -\xi^* + f_i + \mu^*_i$. Then, $(\lambda^*, \mu^*, \rho_i^*)$ is an optimal solution to the dual problem of (MP) with the coupling constraints $-y_i + \sum_{R \in S_i} z_R = 0$, $y_i \geq 0$. Moreover, the reduced cost of $y_i$ is equal to $\rho_i^*$.

*Proof.* After augmenting the coupling constraint to (MP), constraints (30) for $i \in M$ are replaced by $\rho_i + \sum_{k \in R} \lambda_k - \mu_i \leq c^i_R$ for $R \in S^i$. The constraints are equivalent to $\rho_i \leq \min_{R \in S^i} \{-\sum_{k \in R} \lambda_k + \mu_i + c^i_R\}$. Its right-hand side is equal to $-\max_{R \in S^i} \sum_{k \in R} (\lambda_k - c^i_k) + f_i + \mu_i = -\xi^* + f_i + \mu_i$. Also, the dual constraint corresponding to $y_i$ is $-\rho_i \leq 0$, and $\rho_i^*$ is feasible to this constraint. Hence, $(\lambda^*, \mu^*, \rho_i^*)$ is a feasible solution. Because the dual objective function is independent of $\rho_i$, $(\lambda^*, \mu^*, \rho_i^*)$ is an optimal solution to the dual problem of (MP) with the coupling constraints. Thus, $\rho_i^*$ is the reduced cost of $y_i$. □

As a result, variable fixing of $y_i = 0$ is as follows. If $y_i$ is equal to zero with reduced cost $\rho_i^*$, and $Z + \rho^*_i > Z$, then $y_i$ can be fixed to zero.

When $v_i = 0$ in an optimal solution to (MP), or equivalently $y_i = 1$, for some $i \in M$ in an optimal solution to (MP), we assume that the coupling constraints $-v_i - \sum_{R \in S} z_R = -1$, $v_i \geq 0$ are included in (MP). The following proposition shows that the reduced cost of $v_i$ can be calculated without adding the additional explicit coupling constraints to (MP).

**Proposition 6.** Let $(\lambda^*, \mu^*)$ be an optimal solution to (DMP). For given $i \in M$, let $\tau_i^* = \mu_i^*$. Let $\bar{\mu}_k^* = 0$ if $k = i$, and $\bar{\mu}_k^* = \mu_k^*$ otherwise. Then, $(\lambda^*, \bar{\mu}^*, \tau_i^*)$ is an optimal solution to the dual problem of (MP) with the coupling constraints $-v_i - \sum_{R \in S} z_R = -1$, $v_i \geq 0$. Moreover, the reduced cost of $v_i$ is equal to $\tau_i^*$.

*Proof.* After augmenting the coupling constraint to (MP), constraints (30) for $i \in M$ are replaced by $-\bar{\mu}_i + \sum_{k \in R} \lambda_k - \mu_i \leq c^i_R$ for $R \in S^i$. These constraints are feasible for $\tau_i = \tau_i^* = \mu_i^*$ and $\mu_i = \bar{\mu}_i = 0$. Also, the dual constraint corresponding to $v_i$ is $-\tau_i \leq 0$, and $\tau_i^*$ is feasible to this constraint. Hence, $(\lambda^*, \bar{\mu}^*, \tau_i^*)$ is a feasible solution. Because the dual objective function is reconstructed to $\sum_{i \in S} \lambda_i - \sum_{i \in M} \mu_i - \tau_i$, its value is not changed when $(\lambda, \mu, \tau_i) = (\lambda^*, \bar{\mu}^*, \tau_i^*)$. Therefore, $(\lambda^*, \bar{\mu}^*, \tau_i^*)$ is an optimal solution to the dual problem of (MP) with the coupling constraints. Thus, $\tau_i^*$ is the reduced cost of $v_i$. □

As a result, variable fixing of $v_i = 0$, or equivalently $y_i = 1$ is as follows. If $v_i$ is equal to zero with reduced cost $\tau_i^*$, and $Z + \tau_i^* > Z$, then $v_i$ can be fixed to zero, or equivalently $y_i$ can be fixed to one.

We note that fixing $x_{ij}$ to one may also be possible using similar approaches. However, the criterion for fixing seems to be more complicated to find, and we need to handle the situation that $x_{ij} = 1$ and $y_i = 0$ during the branch-and-price procedure. Therefore, we did not try fixing $x_{ij}$ to one in our study.

### 3.5. Other implementation issues

In the procedure of the branch-and-price algorithm, a depth-first search is applied for traversing the search tree. It is known to have relatively low performance compared to a best-first search, but it is useful to find feasible solutions and upper bounds earlier. Primal heuristics have been used widely in the branch-and-price algorithm to find good upper bounds. Although there have been many studies on the heuristics for the SSCFLP, little research has been reported on the heuristics for the robust SSCFLP under demand uncertainty. Therefore, we did not try using primal heuristics for our algorithm. We focused on verifying the effectiveness of the pure branch-and-price algorithm in our study.

Infeasibility of (RMP) is also one of the implementation issues of the algorithm. When (RMP) becomes infeasible, the reason can be that (MP) is infeasible, or there are not enough columns to maintain the feasibility of (RMP). However, it is hard to perceive the exact reason during the column generation procedure. Although there is Farkas pricing (Desroizers and Lübbecke, 2010) to detect whether a master problem is infeasible, it is as hard as optimizing a master problem. To avoid the infeasibility of (RMP) in
Table 1: Problem size of SSCFLP instances.

| Set  | Instances | m     | n     | S/D  |
|------|-----------|-------|-------|------|
| T1-1 | D1-D6 (6) | 10    | 20    | 1.32-1.54 |
| T1-2 | D7-D17 (11) | 15    | 30    | 1.33-3.15 |
| T1-3 | D18-D25 (8) | 20    | 40    | 1.30-3.93 |
| T1-4 | D26-D33 (8) | 20    | 50    | 1.27-4.06 |
| T1-5 | D34-D41 (8) | 30    | 60    | 1.64-5.16 |
| T1-6 | D42-D49 (8) | 30    | 70    | 1.43-3.01 |
| T1-7 | D50-D57 (8) | 30    | 90    | 1.49-3.46 |
| T2-1 | H1-H12 (12) | 10    | 50    | 1.37-2.06 |
| T2-2 | H13-H24 (12) | 20    | 50    | 2.77-3.50 |
| T2-3 | H25-H40 (16) | 30    | 150   | 3.03-6.06 |
| T2-4 | H41-H55 (15) | 10-30 | 70-100 | 1.52-8.28 |
| T2-5 | H56-H71 (16) | 30    | 200   | 1.97-3.95 |
| T3-1 | (10) | 30    | 50    | 3.07-5.88 |
| T3-2 | (10) | 30    | 70    | 3.23-5.93 |
| T3-3 | (10) | 50    | 70    | 3.36-5.71 |
| T3-4 | (10) | 50    | 100   | 2.36-4.43 |
| T4-1 | (10) | 30    | 50    | 5.62-6.36 |
| T4-2 | (10) | 30    | 70    | 3.92-4.49 |
| T4-3 | (10) | 50    | 70    | 6.98-8.12 |
| T4-4 | (10) | 50    | 100   | 4.45-5.26 |

advance, a dummy facility covering all customers with a very high fixed cost can be added. In our algorithm, we set the value of the fixed cost to two times the sum of all costs, i.e., \(2(\sum_{i \in M} \sum_{j \in N} c_{ij} + \sum_{i \in M} f_i)\). The dummy facility has only one binary variable \(z_0\). If the value of \(z_0\) is nonzero after the algorithm solves (RMP) and there is no column having a negative reduced cost, then (RMP) is infeasible, and the algorithm can prune the node.

4. Computational experiments

In this section, we report the performance of the proposed branch-and-price algorithm. We implemented the algorithm using C++ with solvers of linear programming problems (CPLEX 12.9) and binary knapsack problems for the master problem and the subproblem, respectively. Computational results of solving the MIP model (RP2) using CPLEX 12.9 are also provided for comparison with our algorithm. Four different sets of the robust SSCFLP problem test instances are considered; the first two sets consist of benchmark instances used in Delmaire et al. (1999) and Holmberg et al. (1999), respectively, with additional parameters for the maximum deviations, and the last two sets consist of randomly generated instances for detailed analysis.

4.1. Test instances

We consider four different sets of test instances of the robust SSCFLP problem. Among them, the first two benchmark sets are directly taken from the previous literature, and the last two sets are generated for the additional experiments and the simulation experiments. The sizes of test instances are listed in Table 1. In the table, we classify each test set into several subsets depending on the size of the problem. Also, S/D represents the ratio of the total capacity of all facilities over the total demand of customers.

The first set (T1) of 57 test instances (D1-D57) ranging from ten candidate facility locations and 20 customers up to 30 locations and 90 customers were proposed by Delmaire et al. (1999). The second set (T2) of 71 test instances (H1-H71) ranging from ten candidate facility locations and 50 customers up to 30 locations and 200 customers were proposed by Holmberg et al. (1999). Test set (T1) and (T2) are divided into seven types and five types, respectively, depending on the instance size. Let \(U\{a, b\}\) be a random variable which has a discrete uniform distribution in \(\{a, a + 1, \cdots, b\}\). Each maximum deviation \(b_j\) of customer demand is calculated by \(b_j = \lfloor d_j \cdot \sigma_j \rfloor\), where \(\sigma_j\) is taken from \(U\{100, 500\}/1000\). Each degree of robustness \(\Gamma_i\) is fixed to 5.
The third set (T3) is generated based on the data generation scheme in Cornuéjols et al. (1991). Nominal customer demands $d_{ij}$, $j \in N$ and capacities of facilities $s_i$, $i \in M$ are firstly taken from $U\{5, 35\}$ and $U\{10, 160\}$, respectively, and the capacities are expanded by the same factor to adjust the ratio of the sum of capacities to the sum of demands appropriately. Set-up costs and allocation costs are obtained using $f_i = [U\{0, 90\} + U\{100, 110\} \sqrt{|S_j|}$ and $c_{ij} = [10d_i \cdot c_{ij}]$, respectively, where $c_{ij}$ is the Euclidean distance between facility $i$ and customer $j$ placed uniformly at random in a unit square. Maximum deviations $b_j$ of customer demands are decided as T1 and T2 and degree of robustness $\Gamma_i$ varies in $\{3, 5, 7\}$.

The fourth set (T4) is generated based on the data generation scheme in Holmberg et al. (1999). Nominal customer demands $d_{ij}$, capacities of facilities $s_i$, and set-up costs $f_i$ are taken from $U\{10, 50\}$, $U\{100, 500\}$, and $U\{300, 700\}$, respectively. Facilities and customers are placed uniformly at random in a square of size $190 \times 190$. Allocation cost $c_{ij}$ of allocating customer $j$ to facility $i$ is obtained by rounding down the Euclidean distance between them. Compared with instances of T2, the set on the same reference (Holmberg et al. 1999), set-up costs are relatively overvalued to reflect the realistic rates between allocation costs and set-up costs. Parameters $b_j$ and $\Gamma_i$ involved in the demand uncertainty and robustness are also obtained as T3.

The last two sets are designed so that we can examine the characteristics of instances for which our algorithm works well or not. For test set T3 and T4, we considered four different facility, customer pairs, i.e. (30, 50), (30, 70), (50, 70), and (50, 100). For each pair, we generated ten instances, totaling 40 instances for each test set.

### 4.2. Computational results

All computational experiments were performed on an Intel® Core® i5-4670 CPU @ 3.40GHz PC with 24GB RAM. The branch-and-price algorithm was implemented with C++ programming language using Microsoft visual studio 2015, and it used ILOG CPLEX 12.9 for the LP solver of the algorithm. We also compared our result with the branch-and-cut algorithm solving (RP2) using ILOG CPLEX 12.9.

In order to compare the experimental results of our branch-and-price algorithm and CPLEX, we report averaged test values for instances in each test set in Table 2, Table 3, Table 4, and Table 5. We report the number of nodes in the branch-and-bound tree (#node), the number of generated columns (#column), the overall computational time of the algorithm in seconds (time), the time for the master problem (time-m), and the subproblem (time-s), respectively. Also, we report the number of nodes (#node) and the overall computational time in seconds (time) for CPLEX. We set the time limit to 3,600 seconds for both of the branch-and-price and CPLEX. If the algorithm could not find an optimal solution of an instance within the time limit, the instance was not included in computing the average computational time for (time), (time-m), and (time-s) of branch-and-price or (time) of CPLEX, and the number of unsolved instances is reported in the parentheses in the table. However, it was considered for obtaining the other numerical values.

We also compare the tightness of the LP-relaxation bound of (AP) and (RP2). Let $Z_{LP}^{AP}$ and $Z_{LP}^{RP}$ be the optimal or best known objective function value of (AP) and (RP2), respectively, and let $Z_{best}^{AP}$ denote the best known objective value for the problem i.e. the smaller of $Z_{LP}^{AP}$ and $Z_{LP}^{RP}$.
Table 4: Computational results for instances of T3.

| set | \( \gamma \) | Branch-and-price | CPLEX | Branch-and-price | CPLEX |
|-----|---|---|---|---|---|
| T3-1 | 0.1 | 319.6 | 619.4 | 41.0 | 23.4 |
| T3-2 | 0.2 | 123.4 | 146.8 | 28.0 | 13.1 |
| T3-3 | 0.3 | 58.3 | 319.6 | 13.0 | 5.2 |
| T3-4 | 0.4 | 29.8 | 139.6 | 5.3 | 2.2 |

Table 5: Computational results for instances of T4.

| set | \( \gamma \) | Branch-and-price | CPLEX | Branch-and-price | CPLEX |
|-----|---|---|---|---|---|
| T4-1 | 0.1 | 319.6 | 619.4 | 41.0 | 23.4 |
| T4-2 | 0.2 | 123.4 | 146.8 | 28.0 | 13.1 |
| T4-3 | 0.3 | 58.3 | 319.6 | 13.0 | 5.2 |
| T4-4 | 0.4 | 29.8 | 139.6 | 5.3 | 2.2 |

and \( Z_{LP}^{RP} \) be the LP-relaxation bound of (AP), (RP2), and (RP2) with default cutting planes of CPLEX at the root node, respectively. We report the gap between the best known objective function value and the LP-relaxation bound of (AP) i.e. \( (gapBP) = (Z_{best} - Z_{AP}) / Z_{best} \times 100\% \) for each instance. We also report gaps between \( Z_{best} \) and LP-relaxation bound of (RP2) without and with default cutting planes of CPLEX at the root node i.e. \( (gapLP) = (Z_{best} - Z_{LP}^{RP}) / Z_{best} \times 100\% \) and \( (gapBC) = (Z_{best} - Z_{RP}^{RP}) / Z_{best} \times 100\% \) for comparison, respectively. The values are averaged and reported for each set of instances, and one problem in T1-3 which does not have a feasible solution was not included in calculating the average values of \( (gapBP) \), \( (gapLP) \), and \( (gapBC) \), respectively.

Table 2 and Table 3 present computational results of the branch-and-price algorithm and CPLEX for the benchmark instances of T1 and T2, respectively. Table 2 illustrates that our branch-and-price algorithm outperforms CPLEX by a wide margin for the instances of T1. The algorithm of CPLEX could obtain optimal solutions for only ten instances out of 57 instances, while our algorithm could find optimal solutions for all instances. Moreover, CPLEX could find an optimal solution faster than our algorithm for only one instance. Also, among the 15 instances unsolved within 3,600 seconds by our algorithm, CPLEX could find better solutions than our algorithm for only three instances. Overall, the branch-and-price algorithm is better than CPLEX for 53 instances out of 57 instances.

However, the computational results for the instances of T2 in Table 3 show the opposite results, unlike the first ones. Our algorithm is better than CPLEX in terms of solving time for only 22 out of 71 instances, although our branch-and-price algorithm obtains optimal solutions for two of the three instances, which can not be exactly solved by CPLEX within 3,600 seconds.
the LP-relaxation of (AP), and such tendency may grow as we solve larger problems.

Table 4 and Table 5 present computational results of the branch-and-price algorithm and CPLEX for the randomly generated instances of T3 and T4, respectively. In total, there are 77 and 58 instances of T3 and T4, respectively, that our algorithm outperforms CPLEX in terms of the computational time and the quality of feasible solutions. For each gamma value, our branch and price algorithm outperforms CPLEX when \( \Gamma_i = 5 \) (19 and 26 instances of T3 and T4, respectively) and \( \Gamma_i = 7 \) (27 and 29 instances of T3 and T4, respectively), and our algorithm slightly underperforms CPLEX when \( \Gamma_i = 3 \) (12 and 23 instances of T3 and T4, respectively).

Moreover, our branch and price algorithm could solve 106 out of 120 instances of T3 and all 120 out of 120 instances of T4, while CPLEX could solve 57 instances of T3 and 108 instances of T4 within 3,600 seconds. In the case of the instances of T3, our branch and price algorithm could solve 34(\( \Gamma_i = 3 \)), 34(\( \Gamma_i = 5 \)), and 38(\( \Gamma_i = 7 \)) instances of T3, but CPLEX could solve 25(\( \Gamma_i = 3 \)), 17(\( \Gamma_i = 5 \)), and 15(\( \Gamma_i = 7 \)). It shows that our algorithm maintains almost the same performance, but the performance of CPLEX decreases significantly when the value of gamma increases. In conclusion, our algorithm has solved the problems that CPLEX could not easily solve, although CPLEX showed better performance in terms of the computational time for some instances, mostly small-sized ones.

This difference in performance can be due to the gaps between the best known objective function value and the LP-relaxation bound. The gap for (AP) seems to be almost independent of the value of gamma, while the gap for (RP2) tends to grow proportional to the value of gamma.

5. Simulation experiments for evaluation of robust optimal solutions

In this section, we report the results of simulation experiments to evaluate the robustness of the solutions of the robust SSCFLP. Trade-off between the robustness of the solutions and additional costs incurred is verified. It illustrates that the robust SSCFLP can deal with the demand uncertainty efficiently with minimal additional costs.

5.1. Design of experiments

For the simulation experiments, We generated two benchmark instances I3 and I4, which have 30 candidate facility locations and 70 customers, like as the instances of T3 and T4 in the previous section, respectively.

We solved each instance of the robust SSCFLP for all \( \sigma_j \in \{0\%, 10\%, 20\%, 30\%, 40\%, 50\%\} \) and \( \Gamma_i \in \{0, 1, 2, 3, 4, 5\} \). The two types of parameters, rate of the maximum possible variations \( \sigma_j \) and degree of robustness \( \Gamma_i \), control the level of robustness for the optimal solutions. When all \( \sigma_j \) and \( \Gamma_i \) are equal to zero, a solution of the original SSCFLP without demand uncertainty is obtained. We compared the robust solutions to the nominal solution in terms of penalty costs, additional available capacities, and the robustness of solutions.

The robustness of solutions was measured by the empirical ratio of infeasibility using the Monte Carlo simulation. In the simulation, the demand of each customer \( j \in N \) is generated from the truncated normal distribution derived from the normal distribution with mean \( d_j \) and standard deviation \( d_j \cdot \Delta \), by cutting off the lower tail under \( d_j \cdot (1 - 2\Delta) \) in the normal distribution, where \( \Delta \) is the level of variability in demands. The lower truncation of normal distribution prevents ridiculously small or negative value of demand. For each demand scenario, the feasibility of a scenario was confirmed by checking whether every opened facility could accommodate the demands of the assigned customers or not. The ratio of infeasibility was obtained by dividing the number of infeasible scenarios by 5,000 demand scenarios. We did the simulation with varying the level of variability in demands \( \Delta \in \{0, 0.05, \cdots, 0.40\} \) for each robust solution.

5.2. Results of experiments and analysis

Figure 1a and Figure 1b illustrate the percentage ratio of infeasibility under the same rate of maximum possible variation \( \sigma_j = 30\% \) for I3 and I4, respectively. It shows that the robustness of solutions depends on the degree of robustness \( \Gamma_i = 0, 1, 2, 3, 4, 5 \) and the rate of demand variation scenario \( \Delta = 0, 0.05, \cdots, 0.40 \).
The ratio of infeasibility becomes smaller as the degree of robustness increases. For example, when the rate of demand variation $\Delta$ is equal to 0.05, the ratio of infeasibility can be improved by 63.6% on I3 and 76.8% on I4 by increasing the value of $\Gamma_i$ from zero to one, respectively. For the better result, we need to pay 0.5% and 1.8% additional penalty costs for I3 and I4 compared to the nominal solution without demand uncertainty, respectively. We also report that the additional penalty costs are 5.3%, 6.4%, 7.5%, and 8.5% for I3 and 3.7%, 5.0%, 6.6%, and 8.0% for I4 when $\Gamma_i = 2, 3, 4, 5$, respectively. Actually, when the rate of demand variation $\Delta$ increases, the degree of robustness $\Gamma_i$ should be higher for improving the rate of feasibility, and it increases the penalty costs. However, we can observe that the penalty costs are not large, compared with the improvement on the robustness of the solutions.

Figure 2a and Figure 2b illustrate the ratio of infeasibility under the same degree of robustness $\Gamma_i = 3$ for I3 and I4, respectively. They show that the ratio of infeasibility depends on the rate of maximum possible variation $\sigma_j = 0, 10, 20, 30, 40, 50$ and the rate of demand variation scenario $\Delta = 0, 0.05, \cdots, 0.40$. The ratio of infeasibility becomes better as the rate of maximum possible variation increases. For example, when the rate of demand variation $\Delta$ is equal to 0.05, the ratio of infeasibility can be improved by 62.8% on I3 and 73.8% on I4 when $\sigma_j$ is increased from zero to ten percents, respectively. It means that the robust solution obtained when $\sigma_j = 10\%$ is much better protected against infeasibility compared to the solution with $\sigma_j = 0\%$ (i.e. nominal problem). For the better result, we pay 0.4% and 1.6% additional penalty costs
for I3 and I4 compared to the nominal solution without demand uncertainty, respectively. We also report that the additional penalty costs are 5.1%, 6.4%, 7.6%, and 8.8% for I3 and 3.5%, 5.0%, 7.1%, and 9.2% for I4, when $\sigma_j = 20\%, 30\%, 40\%, 50\%$, respectively. A similar phenomenon can be observed when the rate of maximum possible variation $\sigma_j$ is changed, compared with varying the degree of robustness $\Gamma_i$.

Figure 3a and Figure 3b illustrate relationship between the ratio of infeasibility and additional penalty costs compared with the nominal solution without demand uncertainty, i.e., $\Gamma_i = 0$ for I3 and I4, respectively. For each possible pair of $\sigma_i$ and $\Gamma_i$, we obtained a robust solution and evaluated the corresponding ratio of infeasibility and additional penalty cost. Then, we plot the corresponding points in Figure 3a and Figure 3b. They demonstrate that the ratio of infeasibility and the penalty costs are approximately inversely related.

Figure 4a and Figure 4b illustrate relationship between additional total capacities and the penalty costs in comparison with the nominal solution without demand uncertainty, i.e., $\Gamma_i = 0$ for I3 and I4, respectively. A robust solution may need to open additional facilities compared to the nominal solution to cope with uncertain demands. Such additional capacity and additional penalty cost are obtained for each possible pair of $\sigma_j$ and $\Gamma_i$ values, and they are plotted in Figure 4a and Figure 4b. From the linear regression with setting the y-intercept at zero, we can see that the rate of additional total capacities is linearly correlated to the rate of penalty costs. The coefficient of determination $R^2$ is equal to 0.793 and 0.875 for I3 and I4, respectively. From this, we can confirm that the additional costs for robust solutions are directly related to the additional capacities.
6. Conclusion

In this paper, we proposed a branch-and-price algorithm for the robust SSCFLP with the cardinality-constrained demand uncertainty set. The algorithm is based on the allocation-based mathematical model induced by the Dantzig-Wolfe decomposition. The pricing subproblem is the robust binary knapsack problem, which can be solved by solving nominal binary knapsack problems at most \( n \) times. The computational results show that our proposed algorithm can solve practical instances better than CPLEX, which solves the MIP reformulation of the robust SSCFLP. We also verify that the trade-off between the robustness of the solutions and additional costs empirically by Monte-Carlo simulation studies.

Further works may be required to improve the branch-and-price algorithm for the robust SSCFLP, and we suggest some of them. Efficient heuristics for the robust SSCFLP will be helpful as primal heuristics for the branch-and-price algorithm. Additionally, an efficient column management technique may help to reduce the size of the restricted master problem. Lastly, it may be worthwhile to adopt some techniques to improve the convergence speed, like the stabilized column generation. Moreover, considering other uncertainty sets of demands, e.g. polyhedral uncertainty set, ellipsoidal uncertainty set, can be interesting subjects for the robust SSCFLP.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korea government (MSIT) (No. 2019R1F1A1061361)

References

Ahuja, R. K., Orlin, J. B., Pallottino, S., Scarparrá, M. P., and Scutella, M. G. (2004). A multi-exchange heuristic for the single-source capacitated facility location problem. *Management Science*, 50(6), 749-760.

Albarede-Sambola, M., Fernández, E., and Saldanha-da-Gama, F. (2011). The facility location problem with Bernoulli demands. *Omega*, 39(3), 335-345.

Baron, O., Milner, J., and Naseraldin, H. (2011). Facility location: A robust optimization approach. *Production and Operations Management*, 20(5), 772-785.

Baron, O., Berman, O., Fazeli-Zarandi, M. M., and Roshanaei, V. (2019). Almost robust discrete optimization. *European Journal of Operational Research*, 276(2), 451-465.

Barceló, J., and Casanovas, J. (1984). A heuristic Lagrangean algorithm for the capacitated plant location problem. *European Journal of Operational Research*, 15(2), 212-226.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., and Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations Research*, 46(3), 316-329.

Beasley, J. E. (1993). Lagrangean heuristics for location problems. *European Journal of Operational Research*, 65(3), 383-399.

Ben-Tal, A., and Nemirovski, A. (1998). Robust convex optimization. *Mathematics of operations research*, 23(4), 769-805.

Ben-Tal, A., and Nemirovski, A. (2000). Robust solutions of linear programming problems contaminated with uncertain data. *Mathematical programming*, 88(3), 411-424.

Beraldi, P., Bruni, M. E., and Conforti, D. (2004). Designing robust emergency medical service via stochastic programming. *European Journal of Operational Research*, 158(1), 184-193.

Bertsimas, D., and Sim, M. (2003). Robust discrete optimization and network flows. *Mathematical Programming*, 98(1), 49-71.

Bertsimas, D., and Sim, M. (2004). The price of robustness. *Operations Research*, 52(1), 35-53.

Bieniek, M. (2015). A note on the facility location problem with stochastic demands. *Omega*, 55, 53-60.

Ceselli, A., and Righini, G. (2005). A branch-and-price algorithm for the capacitated p-median problem. *Networks*, 45(3), 125-142.

Chen, C. H., and Ting, C. J. (2008). Combining lagrangian heuristic and ant colony system to solve the single source capacitated facility location problem. *Transportation Research part E: Logistics and Transportation Review*, 44(6), 1099-1122.

Conteras, I. A., and Díaz, J. A. (2008). Scatter search for the single source capacitated facility location problem. *Annals of Operations Research*, 157(1), 73-89.

Cormuéjols, G., Sridharan, R., and Thizy, J. M. (1991). A comparison of heuristics and relaxations for the capacitated plant location problem. *European Journal of Operational Research*, 50(3), 280-297.

Cortí, M. J., and Captivo, M. E. (2003). Upper and lower bounds for the single source capacitated location problem. *European Journal of Operational Research*, 151(2), 333-351.

Díaz, J. A., and Fernández, E. (2002). A branch-and-price algorithm for the single source capacitated plant location problem. *Journal of the Operational Research Society*, 53(7), 728-740.

De Aragao, M. P., and Uchoa, E. (2003). Integer program reformulation for robust branch-and-cut-and-price algorithms. In *Mathematical program in rio: a conference in honour of nelson maculan* (pp. 56-61).
Delmaire, H., Díaz, J. A., Fernández, E., and Ortega, M. (1999). Reactive GRASP and tabu search based heuristics for the single source capacitated plant location problem. \textit{INFOR: Information Systems and Operational Research}, 37(3), 194-225.

Desrosiers, J., and Lübbecke, M. E. (2010). Branch-price-and-cut algorithm. \textit{Wiley encyclopedia of operations research and management science}.

Fukasawa, R., Longo, H., Lysgaard, J., De Aragão, M. P., Reis, M., Uchoa, E., and Werneck, R. F. (2006). Robust branch-and-cut-and-price for the capacitated vehicle routing problem. \textit{Mathematical programming}, 106(3), 491-511.

Gadegaard, S. L., Klose, A., and Nielsen, L. R. (2018). An improved cut-and-solve algorithm for the single-source capacitated facility location problem. \textit{EURO Journal on Computational Optimization}, 6(1), 1-27.

Gamrath, G. (2010). Generic branch-cut-and-price.

Guastaroba, G., and Speranza, M. G. (2014). A heuristic for BILP problems: the single source capacitated facility location problem. \textit{European Journal of Operational Research}, 238(2), 438-450.

Güler, N., Pachamanova, D., and Çanakoglu, E. (2013). Robust strategies for facility location under uncertainty. \textit{European Journal of Operational Research}, 225(1), 21-35.

Hindi, K. S., and Pińkosz, K. (1999). Efficient solution of large scale, single-source, capacitated plant location problems. \textit{Journal of the Operational Research Society}, 50(3), 268-274.

Holmberg, K., Rönqvist, M., and Yuan, D. (1999). An exact algorithm for the capacitated facility location problems with single sourcing. \textit{European Journal of Operational Research}, 113(3), 544-559.

Klincewicz, J. G., and Luss, H. (1986). A Lagrangian relaxation heuristic for capacitated facility location with single-source constraints. \textit{Journal of the Operational Research Society}, 37(5), 495-500.

Klose, A., and Görtz, S. (2007). A branch-and-price algorithm for the capacitated facility location problem. \textit{European Journal of Operational Research}, 179(3), 1109-1125.

Lee, C., Lee, K., Park, K., and Park, S. (2012). Technical note-branch-and-price-and-cut approach to the robust network design problem without flow bifurcations. \textit{Operations Research}, 60(3), 604-610.

Lee, C., Lee, K., and Park, S. (2012). Robust vehicle routing problem with deadlines and travel time/demand uncertainty. \textit{Journal of the Operational Research Society}, 63(9), 1294-1306.

Lin, C. K. Y. (2009). Stochastic single-source capacitated facility location model with service level requirements. \textit{International Journal of Production Economics}, 117(2), 439-451.

Martello, S., Pisinger, D., and Toth, P. (1999). Dynamic programming and strong bounds for the 0-1 knapsack problem. \textit{Management Science}, 45(3), 414-424.

Monaci, M., Pferschy, U., and Serafini, P. (2013). Exact solution of the robust knapsack problem. \textit{Computers & Operations Research}, 40(11), 2625-2631.

Neebe, A. W., and Rao, M. R. (1983). An algorithm for the fixed-charge assigning users to sources problem. \textit{Journal of the Operational Research Society}, 34(11), 1107-1113.

Owen, S. H., and Daskin, M. S. (1998). Strategic facility location: A review. \textit{European journal of operational research}, 111(3), 423-447.

Pirkul, H. (1987). Efficient algorithms for the capacitated concentrator location problem. \textit{Computers & Operations Research}, 14(3), 197-206.

Pisinger, D. (1997). A minimal algorithm for the 0-1 knapsack problem. \textit{Operations Research}, 45(5), 758-767.

Rönqvist, M., Tragantalerngsak, S., and Holt, J. (1999). A repeated matching heuristic for the single-source capacitated facility location problem. \textit{European Journal of Operational Research}, 116(1), 51-68.

Savelsbergh, M. (1997). A branch-and-price algorithm for the generalized assignment problem. \textit{Operations Research}, 45(6), 831-841.

Snyder, L. V. (2006). Facility location under uncertainty: a review. \textit{IIE Transactions}, 38(7), 547-564.

Snyder, L. V., and Daskin, M. S. (2006). Stochastic p-robust location problems. \textit{IIE Transactions}, 38(11), 971-985.

Soyster, A. L. (1973). Convex programming with set-inclusive constraints and applications to inexact linear programming \textit{Operations Research}, 21(5), 1154-1157.

Sridharan, R. (1993). A Lagrangian heuristic for the capacitated plant location problem with single source constraints. \textit{European Journal of Operational Research}, 66(3), 305-312.

Tragantalerngsak, S., Holt, J., and Rönqvist, M. (2000). An exact method for the two-echelon, single-source, capacitated facility location problem. \textit{European Journal of Operational Research}, 123(3), 473-489.

Yang, Z., Chu, F., and Chen, H. (2012). A cut-and-solve based algorithm for the single-source capacitated facility location problem. \textit{European Journal of Operational Research}, 221(3), 521-532.