CANADA EXCELLENCE RESEARCH CHAIR

DATA SCIENCE FOR REAL-TIME DECISION-MAKING

STOCHASTIC SINGLE-ALLOCATION HUB LOCATION

Borzou Rostami
Nicolas Kämmerling
Christoph Buchheim
Joe Naoum-Sawaya
Uwe Clausen

September 2018

DS4DM-2018-005

POLYTECHNIQUE MONTRÉAL
DÉPARTEMENT DE MATHÉMATIQUES ET GÉNIE INDUSTRIEL
Pavillon André-Aisenstadt
Succursale Centre-Ville C.P. 6079
Montréal - Québec
H3C 3A7 - Canada
Téléphone: 514-340-5121 # 3314

Stochastic Single-Allocation Hub Location

Borzou Rostami
Canada Excellence Research Chair in Data-Science for Real-time Decision-Making, Polytechnique Montréal, GERAD and CIRRELT, Canada, bo.rostami@gmail.com

Nicolas Kämmerling
Institute of Transport Logistics, TU Dortmund University, Germany, nicolas.kaemmerling@tu-dortmund.de

Christoph Buchheim
Fakultät für Mathematik, TU Dortmund University, Germany, christoph.buchheim@math.tu-dortmund.de

Joe Naoum-Sawaya
Ivey Business School, Western University, London, Canada, jnaoum-sawaya@ivey.ca

Uwe Clausen
Institute of Transport Logistics, TU Dortmund University, Germany, uwe.clausen@tu-dortmund.de

This paper presents a variation of the single allocation hub location problem under demand uncertainty. Namely, we consider variable allocations, meaning that the allocation of the spokes to the hubs can be altered after the uncertainty is realized. This is in contrast to the fixed allocation that is addressed in the literature where the spokes are allocated to the chosen hubs before the uncertainty is realized. As shown in the paper, the fixed allocation case can be solved as a deterministic problem using the expected values of the random variables. However, the variable allocation model is a two-stage stochastic program that is challenging to solve. An alternative convex mixed-integer nonlinear formulation is presented for the variable allocation and a customized solution approach based on cutting planes is proposed to address the computational challenges. The proposed solution approach is implemented in a branch-and-cut framework where the cut-generating subproblems are solved combinatorially, i.e. without an optimization solver. Extensive computational results on the single allocation hub location problem and two of its variants, the capacitated case and the single allocation p-median problem are presented. The proposed cutting plane approach outperforms the direct solution of the problem using the state-of-the-art solver GUROBI as well the L-shaped decomposition which is a common approach for addressing two-stage stochastic programs with recourse.

Key words: Single allocation hub location, demand uncertainty, stochastic programming, outer approximation, cutting planes

1. Introduction

A transport network with many sources and sinks can be very expensive to operate if all shipments are transported directly from the source locations to the destination locations. To benefit from economies of scale, a number of hubs are often established to act as transshipment nodes that can handle the passing flow at a reduced cost. Hub nodes are used to sort, consolidate, and redistribute flows and their main purpose is to achieve economies of scale. While the construction and operation of hubs and the resulting detours lead to extra costs, the bundling of flows decreases the overall cost of operation. The hub location problem optimizes the location of hubs and the allocation of origin and destination nodes to the selected hubs in order to route the flow from the origin nodes to the corresponding destinations while minimizing the total cost of the network. The hub location problem arises in several important applications including telecommunication systems (Klincewicz 1998), airline services (Jaillet et al. 1996), postal delivery services (Ernst and Krishnamoorthy 1996), and public transportation (Nickel et al. 2001), among several others.

Hub location problems are part of the strategic planning decisions and thus the exact operational data of the network is usually unknown and can only be approximated at the time the network is planned. One main source of uncertainty are the stochastic shipping volumes. As hub locations are planned well in advance of the actual operation of the network, only statistical data about shipment sizes are typically available. The usual approach of using average values makes it difficult to give a correct estimate of the necessary hub sizes and the optimal allocation and flow routing. Thus it is often necessary to include uncertainty when deciding the location of hubs and the allocation of nodes to the hubs.

This paper considers the single allocation hub location problem (SAHLP) with demand uncertainty. The single allocation problem denotes the case where each node is assigned to a single hub. We also distinguish between fixed and variable allocations. For fixed allocation, the assignments of the spokes (i.e. non-hub nodes) to the hubs are considered as part of the strategic decisions and therefore are first-stage decisions and remain fixed when uncertainty is realized. Alternatively, for variable allocation, the assignments of the spokes to the hubs are more flexible and can be adjusted when the uncertainty is realized and thus the allocation decisions are considered as second-stage decisions which is in line with real-world practices where the hubs are chosen before knowing the demand while the allocations are determined/altered when the actual demand is realized. As detailed next, prior work has addressed the fixed allocation case, while in this paper the variable allocation stochastic hub location problem is introduced and a computationally efficient solution approach based on exploiting the problem formulation using cutting planes is proposed.

1.1. Literature review

Several variations of the hub location problem have been discussed in the literature. One such variation is the multiple allocation problem where the flow of the same spoke node can be routed through different hubs, i.e. the node is allocated to multiple hubs (Ernst and Krishnamoorthy 1998, Campbell 1996). Alternatively, the single allocation problem assigns each spoke node to exactly one hub and accordingly routes the flow (O'Kelly 1987, Campbell 1996, Rostami et al. 2016, Meier et al. 2016). Furthermore, each of these variations can be classified as capacitated or uncapacitated depending on the type of capacity restriction. In particular, there can be limitations on the total flow routed on a hub-hub link (Labbé and Yaman 2004) or on the volume of flow into the hub nodes (Ernst and Krishnamoorthy 1999).

The first mathematical model for the deterministic hub location problem was proposed in O'Kelly (1987) as a quadratic integer programming formulation to minimize the total transport cost for a given number of hubs that need to be located (p-Hub Median Problem). Since then, the hub location literature mainly focused on the deterministic hub location problem where all the data are assumed to be known in advance. Among the related publications are Ernst and Krishnamoorthy (1996), Skorin-Kapov et al. (1996), Campbell et al. (2005a,b), Elhedhli and Wu (2010), Contreras et al. (2011c), Contreras et al. (2011a), Carlsson and Jia (2013), and Meier and Clausen (2017) where different models and solution methods are considered. The recent surveys of Alumur and Kara (2008) and Campbell and O'Kelly (2012) provide a comprehensive overview of the various variations and solution approaches of the hub location problem.

While deterministic hub location problems and their variants have been extensively studied, the literature addressing data uncertainty in the context of hub location problems is rather limited. Among the first articles related to stochastic hub location problems is Marianov and Serra (2003) which focuses on airline services as an application and considers the multi-allocation variant of the problem. In that context, hub airports are modeled as M/D/c queuing systems in order to limit the congestion caused by queuing airplanes at an airport. The authors obtained feasible solutions for instances up to 30 nodes using a tabu search heuristic. Yang (2009) proposes a multi-allocation hub location problem in the context of air freight transportation. The problem is modeled as a two-stage stochastic program where the demand and the discount factors that are associated with hub-to-hub links are stochastic. In the first stage of the problem, the hub locations are decided while the flight routes are second stage decisions. The stochastic program is formulated as a mixed-integer program (MIP) and instances based on an air freight market in China and Taiwan are solved for sizes up to 10 airports while uncertainty is captured by a discrete probability distribution involving only three different scenarios. Contreras et al. (2011b) examine the multi-allocation hub location problem under uncertain shipping volume and transportation cost. In case of demand uncertainty, they show that the proposed stochastic formulation for multi-allocation hub location is equivalent to the deterministic problem in which all random variables are replaced by their average values. However when the transportation costs are stochastic, this result does not apply in general since the routing decisions depend on the transport costs in each scenario. The authors present a Monte-Carlo simulation coupled with a Benders decomposition algorithm and solve instances with up to 50 nodes.

For the single allocation variant of the hub location problem, Alumur et al. (2012) proposes two-stage stochastic programs that consider uncertainty in hub costs and in the demands. They consider the fixed allocation case, since the first stage determines both the locations of the hubs and the allocation of the spokes to the hubs. The second-stage then only calculates the routing costs per scenario based on the first-stage allocation decisions. To capture stochasticity, a discrete probability distribution is assumed and the problem is then reformulated as a MIP. Optimal solutions are presented for instances with up to 25 nodes with a discrete probability distribution of five different scenarios. Qin and Gao (2017) considers a stochastic single allocation hub location problem with deterministic fixed costs and uncertain demands. The problem is formulated as a quadratic program where the hub locations and the allocations are first stage decisions, and the routing decisions are in the second stage of the problem. The authors show that the proposed stochastic quadratic program is equivalent to a deterministic quadratic program in the special case of continuous and strictly increasing uncertainty distribution functions and provide results for instances with 10 nodes and 3 scenarios using a genetic algorithm. Uncertainties in single-allocation hub location problems were not only studied in the input data but also in the operation of hubs (see, for instance, Tran et al. 2016, Rostami et al. 2018) for more details.

1.2. Contribution of this paper

The main contributions of this paper are as follows. (a) We show that the stochastic optimization model that is proposed by Alumur et al. (2012) for the fixed allocation SAHLP is equivalent to the expected value program of SAHLP where the expected values of the demands are used. (b) We propose a two-stage stochastic program for the variable allocation SAHLP and two of its variants the capacitated SAHLP (CSAHLP) and the Single Allocation p-Hub Median Problem (SApHMP). (c) We propose an alternative mixed integer nonlinear programming (MINLP) formulation and a customized solution approach based on cutting planes that computationally outperforms the state-of-the-art solvers. (d) We show that the cutting planes can be generated by solving a cut generation subproblem without using an optimization solver. (e) We conduct extensive computational testing using a well-known data set from the literature to assess the performance and value of the proposed fixed allocation models for the three problem variants SAHLP, CSAHLP, and SApHMP.

Outline of this paper. Following this introductory section, Section 2 introduces the deterministic SAHLP while Section 3 presents the stochastic SAHLP and discusses its two variants, the fixed allocation and the variable allocation. The proposed MINLP reformulation and the cutting plane solution approach are presented in Section 4 while the computational results are discussed in Section 5. Finally a conclusion is given in Section 6. The attached Appendix A discusses and evaluates an alternative solution approach based on L-shaped decomposition.

2. The deterministic single allocation hub location problem

To introduce the notation of SAHLP, we first provide the deterministic formulation. We consider a directed graph $G = (N, A)$, where $N = \{1, 2, \ldots, n\}$ corresponds to the set of nodes that denote the origins, destinations, and possible hub locations, and A is set of arcs that indicate possible direct links between the different nodes. Let w_{ij} be the amount of flow to be transported from node i to node j and d_{ij} the distance between two nodes i and j. We denote by $O_i = \sum_{j\in N} w_{ij}$ and $D_i = \sum_{j\in N} w_{ji}$ the total outgoing flow from node i and the total incoming flow to node i, respectively. For each $k \in N$, f_k represents the fixed set-up cost for locating a hub at node k. The cost per unit of flow for each path $i - k - \ell - j$ from an origin node i to a destination node j passing through hubs k and m respectively, is $\chi d_{ik} + \alpha d_{k\ell} + \delta d_{\ell j}$, where χ, α, and δ are the nonnegative collection, transfer, and distribution costs respectively and d_{ik}, $d_{k\ell}$, and $d_{\ell j}$ are the distances between the pair of nodes. Note that given that hub nodes are fully interconnected, every path between an origin and a destination node will contain at least one and at most two hubs. SAHLP consists of selecting a subset of nodes as hubs and assigning the remaining nodes to these hubs such that each spoke node, is assigned to exactly one hub with the objective of minimizing the overall cost of the network.

To formulate SAHLP, the following allocation variables are introduced

$$x_{ik} = \begin{cases} 1 & \text{if node } i \text{ is allocated to a hub located at node } k \\ 0 & \text{otherwise.} \end{cases}$$

In particular for every node k, x_{kk} indicates whether k is a hub ($x_{kk} = 1$) or not ($x_{kk} = 0$). SAHLP can then be formulated as the following binary quadratic program:

$$\begin{aligned} [\text{SAHLP}] : \min \quad & \sum_{k\in N} f_k x_{kk} + \sum_{i\in N}\sum_{k\in N} d_{ik}\left(\chi O_i + \delta D_i\right) x_{ik} + \sum_{i,k,j,\ell\in N} \alpha\, w_{ij} d_{k\ell} x_{ik} x_{j\ell} && (1) \\ \text{s.t.} \quad & \sum_{k\in N} x_{ik} = 1 \qquad i \in N && (2) \\ & x_{ik} \le x_{kk} \qquad i, k \in N && (3) \\ & x_{ik} \in \{0, 1\} \qquad i, k \in N. && (4) \end{aligned}$$

The objective is to minimize the total cost of the network which includes the cost of setting up the hubs, the cost of collection and distribution of items between the spoke nodes and the hubs, and the cost of transfer between the hubs. Constraints (2) indicate that each node i is allocated to precisely one hub (i.e. single allocation) while Constraints (3) enforce that node i is allocated to a node k only if k is selected as a hub node. The binary conditions are enforced by Constraints (4).

Note that the formulation of SAHLP can be extended to the Single Allocation p-Hub Median Problem (SApHMP) which denotes a variation of SAHLP where a fixed number of nodes are required to act as hubs. SApHMP is thus formulated by replacing the fixed set-up costs of opening the hubs in SAHLP by the requirement of opening exactly p hubs, i.e.,

$$[\text{SApHMP}]: \min \quad \sum_{i\in N}\sum_{k\in N} d_{ik}\,(\chi\, O_i + \delta\, D_i)\, x_{ik} + \sum_{i,k,j,\ell\in N} \alpha\, w_{ij} d_{k\ell} x_{ik} x_{j\ell} \tag{5}$$

$$\text{s.t.} \quad \sum_{k\in N} x_{kk} = p \tag{6}$$

$$(2), (3), (4),$$

where Constraint (6) enforces the number of open hubs to be p.

Another variation of SAHLP is the Capacitated Single Allocation Hub Location Problem (CSAHLP) which introduces capacity constraints at the hub nodes. Given Γ_k the capacity of node k if that node is selected as a hub, CSAHLP is formulated as

$$[\text{CSAHLP}]: \min \quad \sum_{k\in N} f_k x_{kk} + \sum_{i\in N}\sum_{k\in N} d_{ik}\,(\chi\, O_i + \delta\, D_i)\, x_{ik} + \sum_{i,k,j,\ell\in N} \alpha\, w_{ij} d_{k\ell} x_{ik} x_{j\ell} \tag{7}$$

$$\text{s.t.} \quad \sum_{i\in N} O_i x_{ik} \le \Gamma_k x_{kk} \qquad k \in N \tag{8}$$

$$(2), (3), (4),$$

where Constraints (8) restrict the incoming flow of hub nodes to its capacity limit.

In order to solve SAHLP and its variants, many approaches have been proposed in the literature to linearize the quadratic objective function. Skorin-Kapov et al. (1996) and Ernst and Krishnamoorthy (1996) proposed two mixed-integer linear programming (MILP) formulations for the problem based on a path and a flow representation, respectively. The path-based formulation of Skorin-Kapov et al. (1996) has $O(|V|^4)$ variables and $O(|V|^3)$ constraints and its linear programming (LP) relaxation was shown to provide tight lower bounds. However, due to the large number of variables and constraints, the path-based formulation can only be solved for instances of relatively small sizes. Alternatively, the flow-based formulation uses only $O(|V|^3)$ variables and $O(|V|^2)$ constraints to linearize the problem and is typically regarded as the most effective formulation. To formulate the flow-based SAHLP model (SAHLP-flow), a new variable $y_{ik\ell}$ is defined as the total amount of flow originating at node i and routed via hubs located at nodes k then ℓ, respectively. SAHLP-flow is formulated as

$$[\text{SAHLP-flow}]{:}\min \quad \sum_{k\in N} f_k x_{kk} + \sum_{i\in N}\sum_{k\in N} d_{ik}\,(\chi\, O_i + \delta\, D_i)\, x_{ik} + \sum_{i\in N}\sum_{k\in N}\sum_{\ell\in N} \alpha\, d_{k\ell} y_{ik\ell}$$

$$\text{s.t.} \quad (2), (3), (4)$$

$$\sum_{\ell\in N} y_{ik\ell} - \sum_{\ell\in N} y_{i\ell k} = O_i x_{ik} - \sum_{j\in N} w_{ij} x_{jk} \quad \forall i, k \tag{9}$$

$$\sum_{\ell\in N} y_{ik\ell} \le O_i x_{ik} \quad \forall i, k \tag{10}$$

$$y_{ik\ell} \ge 0 \quad \forall i, k, m. \tag{11}$$

Similar to SAHLP, the objective function minimizes the hub setup costs, the cost of collection and distribution, and the inter-hub transfer costs. Besides Constraints (2), (3), (4) which are used in SAHLP, Constraints (9) are flow balance constraints while Constraints (10) ensure that a flow is possible from spoke i to hub k only if node i is allocated to hub k (Correia et al. 2010). Finally, Constraints (11) indicate the non-negativity restriction on variables y.

3. The stochastic SAHLP under demand uncertainty

The deterministic formulation that is presented in Section 2 assumes that the volume of the flow between each of the source-sink pairs is known which may be unrealistic in practice. In reality, flow volumes are stochastic and long term deterministic forecasts are unreliable. Thus the stochastic SAHLP models stochasticity in the flow by the use of random variables with realization only after the hubs are selected. The goal is to account for demand uncertainty in the design phase of the hub and spoke network in order to maintain the operational reliability of the network when the actual demand is realized.

To model demand uncertainty, we consider for each $i,j \in N$, a random variable $w_{ij}(\xi)$, $\xi \in \Xi$ (Ξ is the support of ξ) representing the future flow that needs to be sent from node i to node j. Moreover, let $O_i(\xi) = \sum_{j\in N} w_{ij}(\xi)$ and $D_i(\xi) = \sum_{j\in N} w_{ji}(\xi)$ be random variables representing the total outgoing flow from node i and the total incoming flow to node i, respectively. Next, two variations of SAHLP are presented: fixed allocation and variable allocation. The fixed allocation proposed by Alumur et al. (2012) uses a two-stage stochastic program where the first-stage decisions correspond to the location and allocation decisions, while in the second stage problem, the flows are consolidated and routed through the network. We show that the two-stage stochastic program of SAHLP with fixed allocation is equivalent to solving the deterministic equivalent using the expected value of the random variables. Then for SAHLP with variable allocation, we propose a two-stage stochastic program with recourse, where the first-stage decisions correspond to the location of the hubs and the second-stage decisions correspond to the optimal allocation decisions and the routing of the commodities.

3.1. Stochastic SAHLP with fixed allocation

Following Alumur et al. (2012), a flow based formulation of SAHLP is considered where the allocations are first stage decision variables, i.e., the decisions of which nodes are hubs and the allocation of the spoke nodes to the selected hubs is made before knowing the demands. SAHLP with stochastic demand is then formulated as

$$\mathrm{SP}_{\mathbf{f}}: \quad \min \quad \sum_{k\in N} f_k x_{kk} + \mathbb{E}_{\xi}[Q_{\mathbf{f}}(x,\xi)] \tag{12}$$

$$\text{s.t.} \quad (2),(3),(4),$$

where $\mathbb{E}_\xi$ denotes the mathematical expectation with respect to $\xi \in \Xi$ and

$$
\begin{aligned}
Q_{\mathbf{f}}(x,\xi) = \min \quad & \sum_{i\in N}\sum_{k\in N} d_{ik}\left(\chi\, O_i(\xi) + \delta\, D_i(\xi)\right) x_{ik} + \sum_{i\in N}\sum_{k\in N}\sum_{\ell\in N} \alpha\, d_{k\ell} y_{ik\ell} \\
\text{s.t.} \quad & \sum_{\ell} y_{ik\ell} - \sum_{\ell} y_{i\ell k} = O_i(\xi) x_{ik} - \sum_j w_{ij}(\xi) x_{jk} \quad \forall i,k && (13)\\
& \sum_{\ell} y_{ik\ell} \le O_i(\xi) x_{ik} \quad \forall i,k && (14)\\
& y_{ik\ell} \ge 0 \quad \forall i,k,m. && (15)
\end{aligned}
$$

The first stage decision variables x that denote the hub selection and node allocation are included in problem $\text{SP}_{\mathbf{f}}$ while the variables y that denote the flow decisions are second stage decisions and included in problem $Q_{\mathbf{f}}(x,\xi)$. The first term of the objective function (12) represents the total set-up cost for installing the hubs while the second term evaluates the expected collection, transfer, and distribution costs. All the constraints have the same meaning as in SAHLP-flow.

Alumur et al. (2012) assumes that the uncertainty that is associated with demands can be described by a finite set of scenarios each having a known probability. Thus by defining new flow variables y for each scenario, an equivalent deterministic formulation can be obtained and solved. In the following theorem, we show that problem $\text{SP}_{\mathbf{f}}$ regardless of the support Ξ is in fact a deterministic SAHLP in which each random variable $w_{ij}(\xi)$ for each $i,j \in N$ can be replaced by its expected values.

Theorem 1. *The stochastic program $SP_{\mathbf{f}}$ is equivalent to the following expected value program:*

$$
\begin{aligned}
min \quad & \sum_{k\in N} f_k x_{kk} + \sum_{i\in N}\sum_{k\in N} d_{ik}\big(\chi\,\mathbb{E}_\xi[O_i(\xi)] + \delta\,\mathbb{E}_\xi[D_i(\xi)]\big) x_{ik} + \sum_{i\in N}\sum_{k\in N}\sum_{\ell\in N} \alpha\, d_{k\ell} y_{ik\ell} \\
s.t. \quad & (2),(3),(4) \\
& \sum_{\ell} y_{i\ell k} - \sum_{\ell} y_{ik\ell} = \mathbb{E}_\xi[O_i(\xi)] x_{ik} - \sum_j \mathbb{E}_\xi[w_{ij}(\xi)] x_{jk} \quad \forall i,k && (16)\\
& \sum_{\ell} y_{ik\ell} \le \mathbb{E}_\xi[O_i(\xi)] x_{ik} \quad \forall i,k && (17)\\
& y_{ik\ell} \ge 0 \quad \forall i,k,m. && (18)
\end{aligned}
$$

Proof. The proof follows from the definition of the y variables as the total amount of flow originating at a node and routed through the hubs, and the fact that in the second stage problem, variables x (the hub decisions) are fixed. Thus for any i,k,ℓ, if $x_{ik}=0$ then $y_{ik\ell}=0$, i.e. there will be no flow originating at node i and routed via hubs located at nodes k and ℓ. If $x_{ik}=1$, then the total flow originating at node i and routed via hubs located at nodes k and ℓ is equal to $\sum_j w_{ij} x_{j\ell}$. Therefore, for any value of x, the second stage objective value is

$$
\mathbb{E}_\xi[Q_{\mathbf{f}}(x,\xi)] = \sum_{i\in N}\sum_{k\in N} d_{ik}\left(\chi\,\mathbb{E}_\xi[O_i(\xi)] + \delta\,\mathbb{E}_\xi[D_i(\xi)]\right) x_{ik} + \sum_{i\in N}\sum_{k\in N}\sum_{\ell\in N} \alpha\, d_{k\ell}\, x_{ik} \sum_{j\in N} \mathbb{E}_\xi[w_{ij}(\xi)] x_{j\ell}.
$$

This is true since the x variables are fixed, and thus the summation and the expectation can be interchanged. By linearizing the objective function using the y variables, the results follow and the proof is complete.

□

3.2. Stochastic SAHLP with variable allocation

As discussed in Section 3.1, the fixed allocation formulation assumes that the allocation of the spokes to the hubs cannot be changed when the demand is realized. Alternatively, this section considers the variable allocation problem where the hubs are chosen before knowing the actual demand while the allocation is determined when the actual demand is realized. The advantage of taking variable allocations into account is illustrated in two examples shown in Figures 1 and 2. Each subfigure shows the choice of the hubs and the allocation from the spokes to the hubs for an example of the capacitated hub location problem. Figures 1a and 2a show the solution of the fixed allocation for a case with 5 scenarios. For each of these scenarios the individual spoke allocations are displayed in Figures 1b–1f and Figures 2b–2f. resulting in an overall decrease of 2.0% (from 387599 to 379783) for the example in Figure 1 and of 8.7% (from 410325 to 374532) for the example in Figure 2 in the objective function value. We observe in both examples that different hubs are chosen when variable allocation is used compared to fixed allocation.

The stochastic SAHLP with variable allocation is formulated as a two-stage stochastic program with recourse. The first-stage decisions are the location of the hubs to be opened while the second-stage decisions are the optimal allocation of the spoke nodes to the hub nodes as well as the routing of the flows. To formulate the stochastic SAHLP with variable allocation, we distinguish the hub selection variables from the allocation variables and define the binary variables z_k, $k \in N$ to indicate whether a hub is located at node k or not. The problem is then formulated as

$$\begin{aligned}
\mathrm{SP}_{\mathbf{v}}: \quad & \min \quad \sum_{k \in N} f_k z_k + \mathbb{E}_\xi[Q_{\mathbf{v}}(z,\xi)] \\
& \text{s.t.} \quad \sum_{k \in N} z_k \geq 1 && (19)\\
& \qquad\quad z_k \in \{0,1\} \quad k \in N, && (20)
\end{aligned}$$

where $\mathbb{E}_\xi$ denotes the mathematical expectation with respect to $\xi \in \Xi$ and

$$\begin{aligned}
Q_{\mathbf{v}}(z,\xi) = \min \; & \sum_{\substack{i,k \in N \\ i \neq k}} d_{ik} \left(\chi\, O_i(\xi) + \delta\, D_i(\xi)\right) x_{ik} \\
& + \sum_{i,j \in N} \alpha\, w_{ij}(\xi) \Big(d_{ij} z_i z_j + \sum_{\substack{\ell \in N \\ \ell \neq j}} d_{i\ell} z_i x_{j\ell} + \sum_{\substack{k \in N \\ k \neq i}} d_{kj} x_{ik} z_j + \sum_{\substack{k,\ell \in N \\ k \neq i, \ell \neq j}} d_{k\ell} x_{ik} x_{j\ell} \Big)
\end{aligned}$$

$$\text{s.t.} \quad \sum_{\substack{k\in N\\ k\neq i}} x_{ik} = 1 - z_i \quad i \in N \tag{21}$$

$$x_{ik} \leq z_k \quad i,k \in N,\ i \neq k \tag{22}$$

$$x_{ik} \in \{0,1\} \quad i,k \in N,\ i \neq k. \tag{23}$$

Constraint (19) is added to force the opening of at least one hub since in any solution of SAHLP there should be at least one hub. Contrary to the fixed allocation case, $w_{ij}(\xi)$ in $\mathrm{SP}_{\mathbf{v}}$ cannot be substituted by its expected value in order to obtain an equivalent deterministic problem since the optimal solution of the second stage depends on the particular realization of the random variables ξ.

A deterministic equivalent formulation of $\mathrm{SP}_{\mathbf{v}}$ can be obtained by assuming that the random parameter ξ follows a discrete distribution with finite support $S_w = \{s_1, \ldots, s_m\}$ and the corresponding probabilities are $p_{s_1}, \ldots, p_{s_m}$ where $p_s = P(\xi = s)$, $s \in S_w$. Accordingly, for each scenario $s \in S_w$, w^s_{ij} denotes the amount of flow from node i to node j, $O^s_i = \sum_{j\in N} w^s_{ij}$ is the total outgoing flow from node i, and $D^s_i = \sum_{j\in N} w^s_{ji}$ is the total incoming flow to node i. Since the node allocations are decisions that will be taken in the future when scenario s is observed, the x variables are redefined as:

$$x^s_{ik} = \begin{cases} 1 & \text{if a node } i \text{ is allocated to a hub located at node } k \text{ under scenario } s \in S_w \\ 0 & \text{otherwise.} \end{cases}$$

The deterministic equivalent formulation of $\mathrm{SP}_{\mathbf{v}}$ is then stated as:

$$\mathrm{DEF}_{\mathbf{v}}: \quad \min \quad \sum_{k\in N} f_k z_k + \sum_{s\in S_w} p_s \sum_{\substack{i,k\in N\\ i\neq k}} c^s_{ik}\, x^s_{ik} +$$
$$\sum_{s\in S_w} p_s \sum_{i,j\in N} \alpha\, w^s_{ij} \Big(d_{ij} z_i z_j + \sum_{\substack{\ell\in N\\ j\neq \ell}} d_{i\ell} z_i x^s_{j\ell} + \sum_{\substack{k\in N\\ i\neq k}} d_{kj} x^s_{ik} z_j + \sum_{\substack{k,\ell\in N\\ i\neq k, j\neq \ell}} d_{k\ell} x^s_{ik} x^s_{j\ell} \Big)$$

$$\text{s.t.} \quad \sum_{\substack{k\in N\\ i\neq k}} x^s_{ik} = 1 - z_i \quad i \in N, s \in S_w \tag{24}$$

$$x^s_{ik} \leq z_k \quad i,k \in N,\ i \neq k, s \in S_w \tag{25}$$

$$z_i \in \{0,1\} \quad \forall i \in N \tag{26}$$

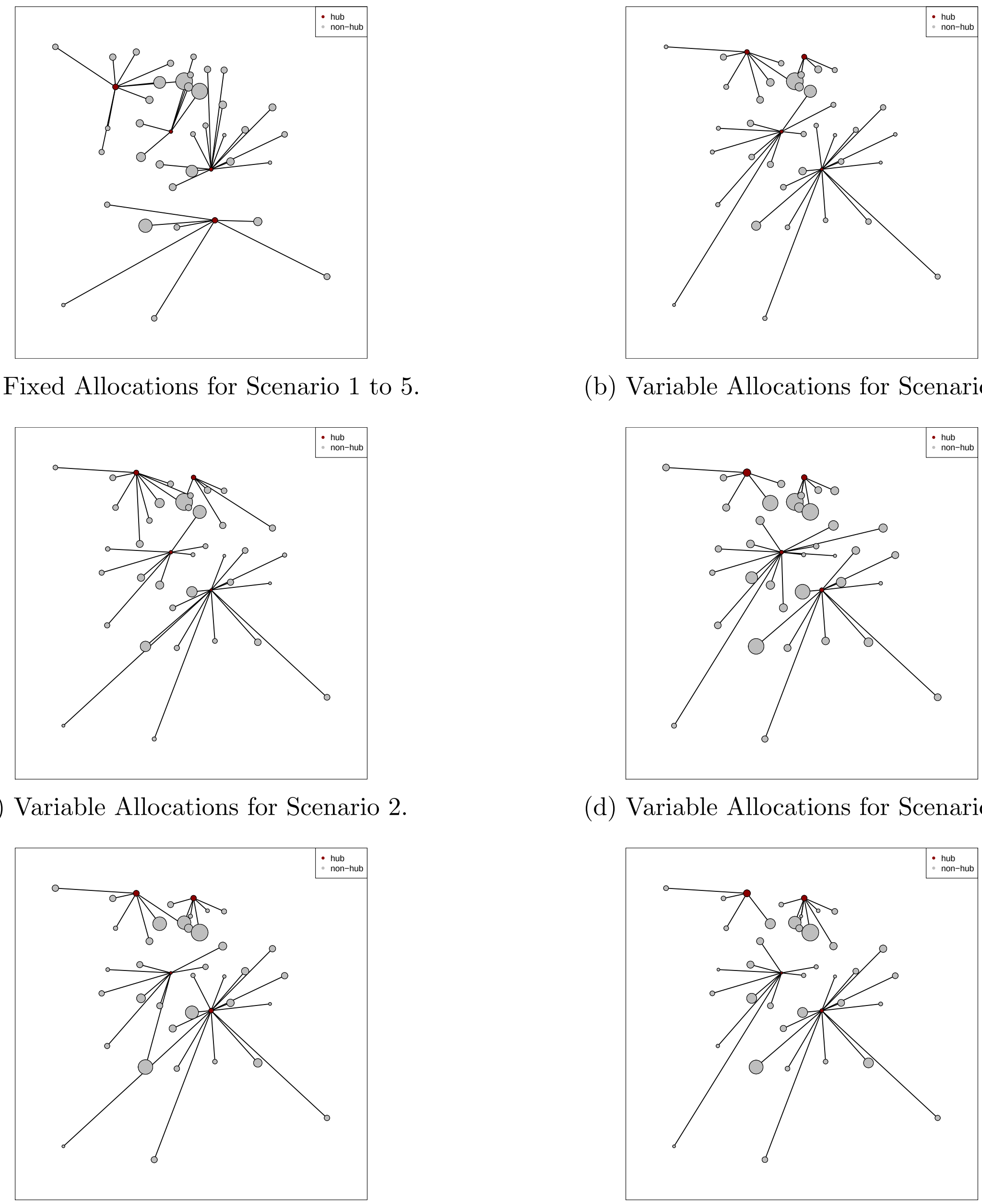
$$x^s_{ik} \in \{0,1\} \quad \forall i,k \in N, s \in S_w. \tag{27}$$

where $c^s_{ik} = d_{ik}\,(\chi\, O^s_i + \delta\, D^s_i)$.

In the following section, an alternative MINLP formulation for the variable allocation SAHLP is proposed and an exact solution approach based on cutting planes is presented.

4. A MINLP reformulation and a cutting plane approach

Due to the quadratic structure of $\text{DEF}_{\mathbf{v}}$, a natural way to tackle this problem is to lineraize it using either the path-based (Skorin-Kapov et al. 1996) or the flow-based (Ernst and Krishnamoorthy 1996) representations and solving the resulting MILPs using a commercial solver. However, it is practically impossible to solve these MILPs for large-size instances in reasonable computational



(a) Fixed Allocations for Scenario 1 to 5.

(b) Variable Allocations for Scenario 1.

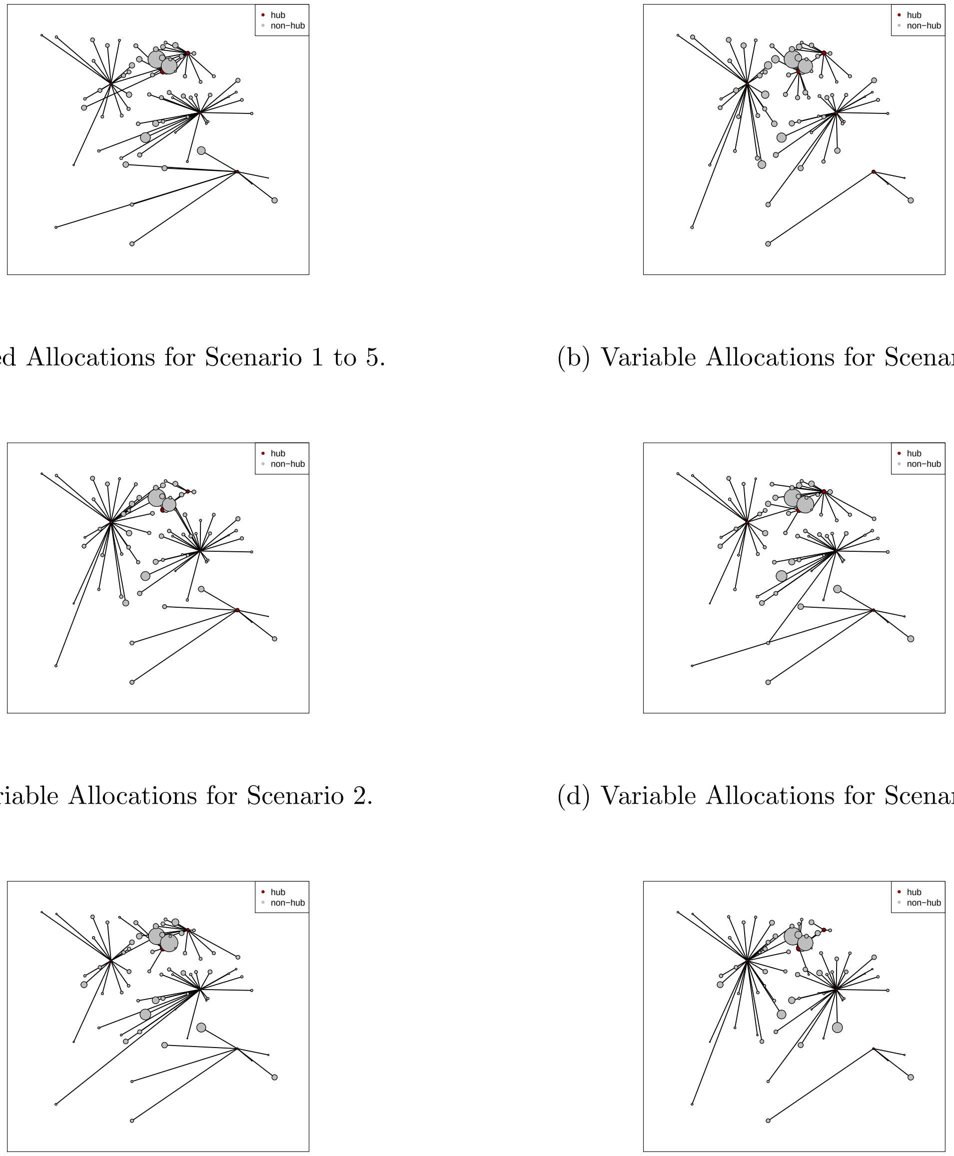
(c) Variable Allocations for Scenario 2.

(d) Variable Allocations for Scenario 3.

(e) Variable Allocations for Scenario 4.

(f) Variable Allocations for Scenario 5.

Figure 1 **Fixed and Variable Allocations for a 40 node instance. The area of each node is proportional to its outgoing flow.**



(a) Fixed Allocations for Scenario 1 to 5.

(b) Variable Allocations for Scenario 1.

(c) Variable Allocations for Scenario 2.

(d) Variable Allocations for Scenario 3.

(e) Variable Allocations for Scenario 4.

(f) Variable Allocations for Scenario 5.

Figure 2 **Fixed and Variable Allocations for a 50 node instance. The area of each node is proportional to its outgoing flow.**

times (see Section 5). The L-shaped decomposition is an alternative approach that is typical applied to solve two-stage stochastic programs with recourse. However, as discussed in Appendix A, its performance is very poor even in solving medium-size instances.

This section thus proposes a MINLP reformulation of problem $\text{DEF}_{\mathbf{v}}$. This new MINLP model has a special structure that can be exploited to develop an effective solution approach as described in Section 4.2.

4.1. MINLP reformulation

The MINLP reformulation of $\mathrm{DEF}_{\mathbf{v}}$ is

$$\begin{aligned}
\mathrm{R}_{\mathbf{v}}:\ \min\quad & \sum_{k\in N} f_k z_k + \sum_{s\in S_w} p_s \sum_{\substack{i,k\in N\\ i\neq k}} c^s_{ik}\, x^s_{ik} + \\
& \sum_{s\in S_w} p_s \sum_{i,j\in N} \alpha\, w^s_{ij}\Big(u^s_{ii} z_i + \sum_{\substack{k\in N\\ k\neq i}} u^s_{ik} x^s_{ik} + v^s_{ij} z_j + \sum_{\substack{\ell\in N\\ \ell\neq j}} v^s_{i\ell} x^s_{j\ell}\Big) && (28)\\
\text{s.t.}\quad & (24)-(27)\\
& u^s_{ik} + v^s_{i\ell} \geq d_{k\ell} \quad i,k,\ell\in N,\ s\in S_w && (29)\\
& u, v \text{ unrestricted}, && (30)
\end{aligned}$$

where the quadratic part of the objective function has been replaced by (28), (29), and (30).

THEOREM 2. *Problem $R_{\mathbf{v}}$ is a reformulation of $DEF_{\mathbf{v}}$.*

Proof. We need to show that for any feasible solution (z,x) of $\mathrm{DEF}_{\mathbf{v}}$, there exists (u,v) such that (z,x,u,v) is feasible for $\mathrm{R}_{\mathbf{v}}$ with the same objective value. Conversely, for any feasible solution (z,x,u,v) of $\mathrm{R}_{\mathbf{v}}$, the corresponding (z,x) is feasible for $\mathrm{DEF}_{\mathbf{v}}$ with the same objective value. For each $s\in S_w$, and $i,k\in N$ let

$$X^s_{ik} = \begin{cases} z_i & \text{if } i=k,\\ x^s_{ik} & \text{if } i\neq k.\end{cases}$$

Given a feasible solution X of the $\mathrm{DEF}_{\mathbf{v}}$, then according to constraints (24), for each $s\in S_w$, there exist two possible cases for each $i\in N$. In the first case, node i is a hub, i.e., $z_i=1$ and $x_{ik}=0$, for all $k\in N$, $k\neq i$. In the second case, node i is not a hub, i.e., there exist a hub $k\neq i$ such that $x_{ik}=1$. Therefore, for each i there exists a $h(i)$ such that for each $s\in S_w$, $X^s_{ih(i)}=1$. Hence, the value of the objective function of $\mathrm{DEF}_{\mathbf{v}}$ is given by

$$\sum_{s\in S_w} p_s \sum_{i\in N} \breve{c}^s_{ih(i)} + \sum_{s\in S_w} p_s \sum_{i,j\in N} \alpha\, w^s_{ij} d_{h(i)h(j)} \tag{31}$$

where $\breve{c}^s_{ih(i)} = c^s_{ih(i)}$ if $h(i)\neq i$, and $\breve{c}^s_{ih(i)} = f_i$ if otherwise.

Furthermore, for each $i,k,\ell\in N$, such that $k=h(i)$ and $\ell=h(j)$ for some $j\in N$ with $X_{jh(j)}=1$, if we set $u_{ik}+v_{i\ell}=d_{k\ell}$, then (X,u,v) is feasible for $\mathrm{R}_{\mathbf{v}}$. The value of the objective function of $\mathrm{R}_{\mathbf{v}}$ is then

$$\begin{aligned}
&\sum_{s\in S_w} p_s \sum_{i\in N} \breve{c}^s_{ih(i)} + \sum_{s\in S_w} p_s \sum_{i,j\in N} \alpha\, w^s_{ij}\Big(\sum_{k\in N} u^s_{ik} X^s_{ik} + \sum_{\ell\in N} v^s_{i\ell} X^s_{j\ell}\Big) =\\
&\sum_{s\in S_w} p_s \sum_{i\in N} \breve{c}^s_{ih(i)} + \sum_{s\in S_w} p_s \sum_{i,j\in N} \alpha\, w^s_{ij}(u_{ih(i)} + v_{ih(j)}).
\end{aligned}$$

which is identical to (31).

Conversely, consider a feasible solution (X,u,v) of $\mathrm{R}_{\mathbf{v}}$ where the inequalities (29) are tight. Indeed, because of the sign of the objective function, for any feasible solution X there exists a (u,v) for which the inequalities are tight. In this case X is also feasible solution for $\mathrm{DEF}_{\mathbf{v}}$. It remains to show that both objective values are also identical. This done by the same calculation as above, just in reverse order.

□

By projecting out the variables u and v, $\mathrm{R}_{\mathbf{v}}$ can be rewritten as

$$\begin{aligned}\mathrm{MP}_{\mathbf{v}}:\ \min\ & \sum_{k\in N} f_k z_k + \sum_{s\in S_w}\sum_{\substack{i,k\in N\\ i\neq k}} p_s\, c^s_{ik}\, x^s_{ik} + \sum_{s\in S_w}\sum_{i\in N} p_s \alpha\, \eta^s_i \\ \text{s.t.}\quad & \eta^s_i \geq \psi^s_i(z,x) \quad i\in N,\ s\in S_w \\ & (24)-(27),\end{aligned}$$

where for each $i\in N$, and $s\in S_w$

$$\begin{aligned}\psi^s_i(\bar z,\bar x) = \max\quad & \sum_{j\in N} w^s_{ij}\Big(u^s_{ii}\bar z_i + \sum_{\substack{k\in N\\ k\neq i}} u^s_{ik}\bar x^s_{ik} + v^s_{ij}\bar z_j + \sum_{\substack{\ell\in N\\ \ell\neq j}} v^s_{i\ell}\bar x^s_{j\ell}\Big) \\ \text{s.t.}\quad & u^s_{ik} + v^s_{i\ell} \leq d_{k\ell} \quad k,\ell\in N & (32)\\ & u,v \text{ unrestricted.} & (33)\end{aligned}$$

Note that since the function $\psi^s_i(\bar z,\bar x)$ for each $i\in N$, and $s\in S_w$ is a point-wise maximum of linear functions of $(\bar z,\bar x)$ then problem $\mathrm{MP}_{\mathbf{v}}$ is a convex MINLP reformulation of $\mathrm{DEF}_{\mathbf{v}}$.

4.2. Exact solution approach based on outer approximation and cutting planes

Since $\mathrm{MP}_{\mathbf{v}}$ is convex, and due to the fact that its objective function is linear, then the optimal solution of $\mathrm{MP}_{\mathbf{v}}$ always lies on the boundary of the convex hull of the feasible set and therefore a cutting-plane approach can be used to solve the problem to optimality. More precisely, for a feasible solution $\bar X$ such that

$$\bar X^s_{ik} = \begin{cases}\bar z_i & \text{if } i=k,\\ \bar x^s_{ik} & \text{if } i\neq k,\end{cases}$$

$\psi^s_i(\bar z,\bar x)$ can be approximated by a supporting hyperplane at $\bar X$. Particularly, given $\pi^s_i \in \partial\psi^s_i(\bar z,\bar x)$ the subgradient of $\psi^s_i(\bar z,\bar x)$ at $\bar X$ and following the approach of Duran and Grossmann (1986), $\psi^s_i(\bar z,\bar x)$ can be approximated at $\bar X$ using the following subgradient cuts

$$\eta^s_i \geq \psi^s_i(z,x) \geq \psi^s_i(\bar X^s) + \pi^s_i(X^s - \bar X^s) \quad i\in N,\ s\in S_w. \tag{34}$$

Thus $\mathrm{MP}_{\mathbf{v}}$ can be solved as a mixed integer linear program where cuts (34) are generated in a branch-and-cut framework.

Given a feasible solution $\bar{X} \in [0,1]$ for each $s \in S_w$ and $i \in N$, the cut generating subproblem is

$$\begin{aligned}
\mathrm{PS(s,}\ i,\ \bar{X}^s\mathrm{):}\quad \max \quad & \sum_{j\in N} w_{ij}^s \big(\sum_{k\in N} \bar{X}_{ik}^s u_{ik}^s + \sum_{\ell\in N} \bar{X}_{j\ell}^s v_{i\ell}^s \big) \\
\text{s.t.} \quad & u_{ik}^s + v_{i\ell}^s \le d_{k\ell} \quad k,\ell \in N && (35)\\
& u_{ik}^s, v_{i\ell}^s \in \mathbb{R} \quad \forall k,\ell \in N. && (36)
\end{aligned}$$

We note that PS(s, i, $\bar{X}^s$) is a linear program, and hence can be solved efficiently using a state-of-the-art optimization solver. However, as shown by the following theorem, the structure of PS(s, i, $\bar{X}^s$) can be exploited to obtain an optimal solution $(\bar{u}, \bar{v})$ more efficiently compared to using a linear programming solver.

Theorem 3. *Given a solution $\bar{X} \in \{0,1\}$, that satisfys constraints* (24) *and* (25)*, an optimal solution of subproblem PS(s,i,$\bar{X}$) can be obtained by setting*

$$\bar{v}_{i\ell}^s = \sum_{k\in N} d_{k\ell} \bar{X}_{ik}^s \quad \forall \ell \in N \tag{37}$$

$$\bar{u}_{ik}^s = \min_{\ell\in N} \big\{ d_{k\ell} - \bar{v}_{i\ell}^s \big\} \quad \forall k \in N. \tag{38}$$

Proof. The solution $(\bar{u}, \bar{v})$ is feasible for PS(s,i,$\bar{X}^s$), i.e. for all $k,\ell \in N$

$$\begin{aligned}
u_{ik}^s + v_{i\ell}^s &= \sum_{k\in N} d_{k\ell} \bar{X}_{ik}^s + \min_{\ell\in N} \big\{ d_{kl} - \bar{v}_{i\ell}^s \big\} \\
&\le \sum_{k\in N} d_{k\ell} \bar{X}_{ik}^s + d_{k\ell} - \sum_{k\in N} d_{k\ell} \bar{X}_{ik}^s = d_{k\ell}.
\end{aligned}$$

Given dual variables λ_{ikl}^s of Constraints (35), the dual problem of PS(s,i,$\bar{X}$) is

$$\begin{aligned}
\mathrm{DS(s,}\ i,\ \bar{X}^s\mathrm{):}\quad \min \quad & \sum_{k\in N} \sum_{\ell\in N} d_{k\ell} \lambda_{ik\ell}^s \\
\text{s.t.} \quad & \sum_{k\in N} \lambda_{ik\ell}^s = \sum_{j\in N} w_{ij} X_{j\ell}^s \quad \forall \ell \in N && (39)\\
& \sum_{\ell\in N} \lambda_{ik\ell}^s = \sum_{j\in N} w_{ij} X_{ik}^s \quad \forall k \in N && (40)\\
& \lambda_{ik\ell}^s \ge 0 \quad \forall k,\ell \in N. && (41)
\end{aligned}$$

Setting the dual variables to

$$\bar{\lambda}_{ik\ell}^s = \bar{X}_{ik}^s \sum_{j} w_{ij}^s \bar{X}_{j\ell}^s \quad k,\ell \in N,\ s \in S_w,\ i \in N \tag{42}$$

leads to a feasible solution to DS(s,i,$\bar{X}^s$). By LP-duality, it then suffices to show that the objective value of PS(s,i,$\bar{X}$) at $(\bar{u},\bar{v})$ matches the objective value of DS(s,i,$\bar{X}^s$) at $\bar{\lambda}$, i.e.,

$$\begin{aligned}
&\sum_{j\in N} w_{ij}^s \big(\sum_{k\in N} \bar{X}_{ik}^s \bar{u}_{ik}^s + \sum_{\ell\in N} \bar{X}_{j\ell}^s \bar{v}_{i\ell}^s \big) \\
&= \sum_{k\in N} O_i^s \bar{X}_{ik}^s \bar{u}_{ik}^s + \sum_{j,\ell\in N} w_{ij}^s \bar{X}_{j\ell}^s \bar{v}_{i\ell}^s \\
&= \sum_{k\in N} O_i^s \bar{X}_{ik}^s (d_{k,a(k)} - \sum_{\ell\in N} d_{k,a(k)} \bar{X}_{i\ell}^s) + \sum_{j,\ell\in N} w_{ij}^s \bar{X}_{j\ell}^s (\sum_{k\in N} d_{k\ell} \bar{X}_{ik}^s) \\
&= \sum_{k\in N} O_i^s \, d_{k,a(k)} \, \bar{X}_{ik}^s - \sum_{k,\ell\in N} O_i^s \, d_{\ell,a(k)} \, \bar{X}_{ik}^s \bar{X}_{i\ell}^s + \sum_{j,\ell\in N} w_{ij}^s \bar{X}_{j\ell}^s (\sum_{k\in N} d_{k\ell} \bar{X}_{ik}^s) \\
&= \sum_{k,\ell\in N} d_{k\ell} \bar{X}_{ik}^s \sum_{j\in N} w_{ij}^s \bar{X}_{j\ell}^s = \sum_{k,\ell\in N} d_{k\ell} \bar{\lambda}_{ik\ell}^s
\end{aligned}$$

where for each $k \in N$, $a(k) = \arg\min_{\ell\in N} \{d_{k\ell} - \bar{v}_{i\ell}^s\}$.

□

Thus following Theorem 3, the optimal solution $\bar{u}_{ik}$ and $\bar{v}_{i\ell}$ of PS(s,i,$\bar{X}$) and the corresponding subgradient cut

$$\eta_i^s \geq \sum_{i,j\in N} w_{ij}^s \Big(\bar{u}_{ii}^s z_i + \sum_{\substack{k\in N\\ k\neq i}} \bar{u}_{ik}^s x_{ik}^s + \bar{v}_{ij}^s z_j + \sum_{\substack{\ell\in N\\ \ell\neq j}} \bar{v}_{i\ell}^s x_{j\ell}^s \Big) \tag{43}$$

are generated without the need of an optimization solver. Since the state-of-the-art optimization solvers provide a branch-and-cut framework supported by the use of callbacks, a branch-and-cut approach is adopted to generate cuts (43). In our implementation which is evaluated in the following section, the violated cuts (43) are added at the root node and at nodes where a candidate incumbent is available.

Finally we note that, while the focus throughout this paper is on demand uncertainty, the proposed solution approach is independent of demand uncertainty and can also be applied to solve problems with uncertainty in allocation costs and flow routing costs which are modelled as two-stage stochastic programs with recourse.

5. Computational results

This section provides computational experiments to evaluate the proposed cutting plane approach in solving the stochastic single allocation hub location problem (SAHLP) along with its variants: the single allocation p-hub median problem (SApHMP) and the capacitated SAHLP problem (CSAHLP). Details regarding the implementation and the test instances are provided first then the computational results are presented.

5.1. Implementation and test instances

The proposed approach is implemented in C++ using GUROBI 7.0 callback framework and the results are compared to the direct solution of the deterministic equivalent problem (problem $\text{DEF}_{\mathbf{v}}$ and its CSHALP and SApHMP variants) using GUROBI with default settings. All the experiments are conducted using a single Intel Xeon E5-1630v3 (3.7 GHz) processor with 16 gigabytes of RAM and the computational CPU time is limited to 7200 seconds (two hours) for each test instance.

The test instances that are used are the well known AP instances that are commonly used in the literature and can be obtained from the OR Library (Beasley 2012). These test instances which were introduced by Ernst and Krishnamoorthy (1996) are based on the mail flow of the Australian post. The transportation cost parameters are set to $\alpha = 0.75$, $\chi = 3.0$, and $\delta = 2.0$ as specified in the AP dataset.

5.2. Scenario generation

To generate the different flow scenarios for each source-sink pair, we make the reasonable assumption that the flow takes the form of a discrete unit (ex: number of parcels, number of pallets, etc.) and is generated by independent customers. Particularly, we assume that there are C customers that are sending goods between two particular nodes where each customer $c \in C$ sends a single unit in a day with a probability p_c. According to these assumptions and as shown in Harremoës (2001), a Poisson distribution is the most appropriate distribution to model the flow between two nodes (the Poisson distribution provides the maximum entropy). Since some nodes might have high or low demands, we also consider for each node $i \in N$ a multiplicative factor π_i which denotes the deviation from the base case and assume that π_i is uniformly distributed in the interval $[0.5,\ 1.5]$. Thus for every source-sink pair (i,j), we assume that the amount of flow is given by a probability distribution $\mathcal{P}(\pi_i \pi_j w_{ij})$ on $\mathbb{Z}_{\geq 0}$, where $\mathcal{P}(\lambda)$ denotes the Poisson distribution with expected value λ.

5.3. Computational evaluation of the cutting plane approach

This section provides detailed computational results that illustrate the performance of the proposed cutting plane in solving SAHLP, SApHMP, and CSAHLP. The proposed cutting plane approach is compared to the direct solution of the deterministic equivalent formulations using GUROBI. In all the computational test that are presented in this section, 5 scenarios generated randomly as discussed in Section 5.2 are considered. The following details are reported in Tables 1–7:

N : Number of nodes in the network.
$Type$: Type of the instance (applies only to SAHLP and CSHALP)
p : Desired number of hubs (applies only to SApHMP).
$CUTS0$: Number of cuts that are generated by the cutting plane approach at the root node.
$CPU0$: Computational time spent on generating cuts at the root node.
$CUTS$: Total number of cuts generated in the branch-and-cut tree.
$Nodes$: Number of nodes explored in the branch-and-cut tree.
CPU : Total computational time.
Opt : Objective function value.

Two types of instances denoted by L and T are tested for SAHLP. As detailed in Ernst and Krishnamoorthy (1999), the instances with type T have higher fixed costs for the nodes with large flows while the instances of type L do not exhibit this trend. For CSHALP, four types of instances denoted by LL, LT, TL, and TT are tested (Additional details about the characteristics of the instances can be found in Ernst and Krishnamoorthy (1999)). The first letter denotes the fixed cost type similar to SAHLP while the second letter indicates tight (T) and loose (L) node capacities. In the reported results, all CPU times are given in seconds. The optimal objective function values are displayed for the cases that were solved within the time limit whereas the best lower and upper bounds are displayed along with the resulting percentage gap for the cases where the time limit was exceeded. The instances that are marked by bold indicate the approach that outperforms the others in terms of computational time if the problem is solved within the time limit or in terms of the optimality gap if the time limit is reached.

5.3.1. Single allocation hub location problem The results for SAHLP are reported in Table 1. As shown, the proposed cutting plane approach is capable of solving the problem to optimality up to sizes of 200 nodes within the two hours time limit while GUROBI reaches the time limit starting with 125 nodes for the (L) instances and 175 nodes for the (T) instances. The (L) instances appear to be more challenging to solve than the (T) instances for both the cutting plane approach and GUROBI. For the cutting plane approach, all the (T) instances are solved in lower computational time than the (L) instances. The results also show that the majority of the cuts are generated at the root node. On average 70% of the cuts are generated at the root node while the remaining cuts are generated at the other nodes of the tree. For the cutting plane approach, the total number of nodes that are explored is significantly higher than GUROBI where several instances are solved to optimality at the root node or within very few nodes. However this comes at the expense of the total computational time where the proposed cutting plane approach solves the problems at a fraction of the time that is taken GUROBI. The geometric mean of the ratio of the computational times shows that GUROBI is on average 20 times slower than the proposed cutting plane approach.

5.3.2. Single Allocation p-Hub Median Problem Another variant of SHALP is the Single Allocation p-Hub Median Problem. As detailed in Section 2, SApHMP replaces the cost of opening a hub by a constraint that requires exactly p nodes to act as hubs in the network. Table 2 presents the computational performance of the proposed cutting plane algorithm as well as that of GUROBI for solving SApHMP. For each instance, the desired number of hubs p is varied between 2 and 5. The proposed cutting plane algorithm consistently outperforms GUROBI and is able to solve 37 instances of the 44 tested instances to optimality within the two hours time limit. For

Table 1 Computational results for SAHLP.

Instance		Cutting Plane						GUROBI		
N	type	CUTS0	CPU0	CUTS	Nodes	CPU	Opt.	Nodes	CPU	Opt.
25	L	197	0	239	332	**1**	203271	0	8	203271
25	T	125	0	125	0	**0**	240689	0	4	240689
40	L	800	2	1153	171	**5**	257241	5	250	257241
40	T	554	1	930	56	**3**	307257	0	74	307257
50	L	802	3	1105	521	**20**	219345	0	161	219345
50	T	235	0	235	0	**1**	277276	0	120	277276
60	L	625	4	890	521	**17**	230305	0	307	230305
60	T	571	3	1155	1664	**26**	275062	12	459	275062
75	L	1467	7	2258	2418	**236**	271648	40	2059	271648
75	T	1096	7	1259	1034	**21**	325953	0	549	325953
90	L	990	11	2049	714	**339**	244711	3	4029	244711
90	T	1142	11	1400	3	**16**	278762	0	840	278762
100	L	1431	15	1781	1439	**210**	251871	0	2563	251871
100	T	488	11	794	5	**14**	331972	0	1472	331972
125	L	2172	30	3300	3863	**376**	233366	-	$> 7200^*$	-
125	T	605	28	795	522	**47**	262691	0	3002	262691
150	L	1784	44	2137	1584	**589**	241412	-	$> 7200^*$	-
150	T	748	38	960	520	**64**	256235	0	5532	256235
175	L	2000	126	3117	2937	**953**	232238	-	$> 7200^*$	-
175	T	1038	165	2016	780	**323**	251119	-	$> 7200^*$	-
200	L	2275	244	4690	1484	**3555**	251050	-	$> 7200^*$	-
200	T	1823	237	3891	3828	**1892**	260868	-	$> 7200^*$	-
					Geometric Mean:	1			19.89	

-: time limit of 7200 seconds reached at pre-solve stage.
*: 7200 was used when computing the geometric mean.

the instances that were not solved to optimality, the remaining gap between the lower and upper bounds is relatively small ranging between 0.3% and 3.6% for the tested instances. Computing the geometric mean of the ratio of the computational times shows that the proposed cutting plane algorithm is 5 times faster than GUROBI in solving the tested SApHMP instances. GUROBI was able to solve 23 instances to optimality within the 2 hours time limit while for the remaining 21 instances which are mainly with 100 nodes or more, the time limit was reached at the pre-solve stage of the problem.

5.3.3. Capacitated Single Allocation Hub Location Problem As discussed in Section 2, CSAHLP refers to SAHLP with an additional capacity constraint on the hubs. The addition of a capacity constraint is known to significantly complicate the problem as flow is rerouted to accommodate the capacity limits. The problem thus becomes more challenging computationally. The results for CSAHLP which are reported in Table 3 show that the problem is more computationally challenging than SAHLP with the proposed cutting plane approach being able to solve 24 of the 44 instances to optimality within the two hours time limit while GUROBI is capable of solving only 17. Similar to SAHLP and SApHMP, the majority of the cuts are generated at the root node of the branch-and-cut tree. In terms of computational time, the geometric mean of the computational times shows that the proposed cutting plane algorithm is 5 times faster than GUROBI in solving the tested CSAHLP instances. We note though that GUROBI was able to solve 5 instances in

Table 2 Computational results for SApHMP.

Instance		Cutting Plane						GUROBI		
N	p	CUTS0	CPU0	CUTS	Nodes	CPU	Opt.	Nodes	CPU	Opt.
25	2	359	0	509	256	**1**	146259	0	7	146259
25	3	317	0	412	1169	**2**	125011	0	8	125011
25	4	344	0	523	1267	**2**	110716	0	16	110716
25	5	410	0	573	459	**1**	97740	0	11	97740
40	2	545	1	796	830	**22**	197565	30	146	197565
40	3	749	1	1361	1821	**66**	175623	22	263	175623
40	4	590	1	1453	1254	**91**	159074	128	475	159074
40	5	566	1	1279	26605	**247**	144945	323	571	144945
50	2	291	2	373	519	**5**	157233	0	132	157233
50	3	468	3	628	396	**8**	139851	0	185	139851
50	4	696	2	927	174	**5**	124972	0	176	124972
50	5	675	2	839	260	**6**	113818	0	171	113818
60	2	722	4	920	558	**23**	181998	0	295	181998
60	3	829	4	1467	1170	**230**	162309	0	663	162309
60	4	876	4	1671	5747	**211**	148832	11	1771	148832
60	5	952	5	1981	2284	**1605**	137135	50	2772	137135
75	2	556	7	1157	732	**156**	217498	3	1587	217498
75	3	1170	9	2150	1112	**166**	192734	3	2566	192734
75	4	1184	8	1905	1608	**180**	175058	23	2969	175058
75	5	1069	6	1663	1325	**120**	159837	0	1270	159837
90	2	651	11	1634	1003	**601**	200327	45	3513	200327
90	3	919	13	2201	2450	**278**	176290	-	$>7200^*$	-
90	4	1432	12	2017	1374	**426**	158724	47	3889	158724
90	5	1589	12	3196	3417	**1174**	149298	-	$>7200^*$	-
100	2	643	13	1052	940	**140**	191461	3	3120	191461
100	3	1633	19	3649	562	$>7200^*$	**(170047, 173101; 1.8%)**	-	$>7200^*$	-
100	4	1635	17	2307	4849	**413**	154798	-	$>7200^*$	-
100	5	1480	15	3320	9513	**1379**	144208	-	$>7200^*$	-
125	2	1498	32	2796	2649	**683**	186248	-	$>7200^*$	-
125	3	2207	39	4811	5275	**1709**	166765	-	$>7200^*$	-
125	4	2255	32	4540	6061	**2352**	153478	-	$>7200^*$	-
125	5	1900	30	4981	10349	**6530**	143680	-	$>7200^*$	-
150	2	1461	38	2347	979	**221**	196615	-	$>7200^*$	-
150	3	2096	47	3920	2068	**1147**	175416	-	$>7200^*$	-
150	4	1225	42	2869	11768	**1068**	158099	-	$>7200^*$	-
150	5	1722	51	3210	55183	**1942**	144802	-	$>7200^*$	-
175	2	1459	274	2086	19584	**3195**	186457	-	$>7200^*$	-
175	3	2643	204	4259	522	$>7200^*$	**(165675, 168716; 1.8%)**	-	$>7200^*$	-
175	4	2334	170	4474	525	$>7200^*$	**(150052, 154176; 2.7%)**	-	$>7200^*$	-
175	5	2412	201	4921	522	$>7200^*$	**(139550, 144827; 3.6%)**	-	$>7200^*$	-
200	2	1843	213	2893	947	**4283**	203691	-	$>7200^*$	-
200	3	2835	225	3577	701	$>7200^*$	**(178223, 178832; 0.3%)**	-	$>7200^*$	-
200	4	2477	243	5599	519	$>7200^*$	**(162189, 165820; 2.2%)**	-	$>7200^*$	-
200	5	2866	264	5492	518	$>7200^*$	**(151382, 156660; 3.4%)**	-	$>7200^*$	-
					Geometric Mean:	1			5.49	

-: time limit of 7200 seconds reached at pre-solve stage.
*: 7200 was used when computing the geometric mean.

less computational time than the cutting plane approach and achieved a less gap for one instance. For the 20 instances that were not solved to optimality by the cutting plane approach, the largest remaining gap is 79% while GUROBI was not able to pass the pre-solve stage for 20 instances.

5.3.4. Summary of the results This section provides a summary of the computational performance of the proposed cutting plane approach compared to the performance of GUROBI in solving SAHLP, SApHMP, and CSAHLP. The performance profiles (Dolan and Moré 2002) shown in Figure 3 were constructed to show, for each problem type, the percentage number of instances

Table 3 Computational results for CSAHLP.

Instance		Cutting Plane						GUROBI		
N	type	CUTS0	CPU0	CUTS	Nodes	CPU	Opt.	Nodes	CPU	Opt.
25	LL	197	0	234	209	**1**	203271	0	8	203271
25	TL	382	0	841	190	**3**	262630	92	46	262639
25	LT	476	0	1210	1077	569	229861	260	**62**	229861
25	TT	464	0	1230	2037	195	289729	2697	**143**	289729
40	LL	521	2	1274	574	504	258881	7	**184**	258881
40	TL	533	1	900	868	**47**	309217	8	104	309217
40	LT	702	1	1611	5440	**778**	288546	3825	3259	288546
40	TT	640	2	1519	219441	> 7200*	**(379266, 379805; 0.1%)**	17674	> 7200*	(378017, 382241; 1.1%)
50	LL	892	3	1071	536	**19**	219852	0	113	219852
50	TL	538	4	921	1286	**63**	296194	273	1157	296194
50	LT	726	5	1765	8403	**1953**	251264	3772	5443	251264
50	TT	647	5	958	331900	6993	374532	8501	**6478**	374413
60	LL	844	5	1689	1431	**168**	229586	11	692	229596
60	TL	885	6	1615	7807	**127**	270405	326	580	270405
60	LT	936	9	2437	85420	> 7200*	**(273282, 274154; 0.3%)**	525	> 7200*	(270138, 286505; 5.7%)
60	TT	832	8	1742	29586	3440	415946	64	**1322**	381352
75	LL	1096	11	2301	575	> 7200*	(276436, 327569, 15.6%)	318	> 7200*	**(279703, 322028; 13.1%)**
75	TL	455	8	665	576	**20**	339693	0	547	339693
75	LT	1105	14	2808	14092	> 7200*	**(302231, 309403; 2.3%)**	-	> 7200*	-
75	TT	926	13	1726	160886	> 7200*	**(428650, 429247; 0.1%)**	413	> 7200*	(424475, 555546; 23.6%)
90	LL	1194	16	1862	2240	**1085**	245927	2	3832	245927
90	TL	1263	21	2173	141336	**1842**	326210	-	> 7200*	-
90	LT	1498	38	2779	2289	> 7200*	**(281331, 306430; 8.2%)**	-	> 7200*	-
90	TT	1306	33	2301	4630	> 7200*	**(422624, 544419; 22.3%)**	-	> 7200*	-
100	LL	1928	23	2894	5727	**1276**	258412	-	> 7200*	-
100	TL	1440	18	2261	42939	**391**	382020	1020	> 7200*	(381692, 382125; 0.1%)
100	LT	1940	42	2470	560	> 7200*	**(270716, 303620; 10.9%)**	-	> 7200*	-
100	TT	1357	54	2249	5106	> 7200*	**(510734, 686823; 25.6%)**	-	> 7200*	-
125	LL	1778	96	2968	6110	**2013**	242470	-	> 7200*	-
125	TL	635	38	793	533	**80**	250400	4	3675	250400
125	LT	2298	109	3173	540	> 7200*	**(260806, 271381; 3.9%)**	-	> 7200*	-
125	TT	1356	98	1470	4464	**700**	304256	-	> 7200*	-
150	LL	2818	138	4317	547	> 7200*	**(228864, 232100; 1.4%)**	-	> 7200*	-
150	TL	3066	132	3877	7858	**1377**	261916	-	> 7200*	-
150	LT	2275	221	3294	541	> 7200*	**(238671, 269319; 11.4%)**	-	> 7200*	-
150	TT	2176	225	3408	685	> 7200*	**(291676, 376356; 22.5%)**	-	> 7200*	-
175	LL	2815	270	6390	540	> 7200*	**(236107, 238449; 1.0%)**	-	> 7200*	-
175	TL	1822	236	2551	2998	**1813**	250192	-	> 7200*	-
175	LT	3269	506	4689	561	> 7200*	**(249586, 254953; 2.1%)**	-	> 7200*	-
175	TT	2187	989	3030	1047	> 7200*	**(287615, 318347; 9.7%)**	-	> 7200*	-
200	LL	3224	1085	4669	516	> 7200*	**(253512, 287265; 11.7%)**	-	> 7200*	-
200	TL	2866	491	4353	519	> 7200*	**(262228, 1247994; 79.0%)**	-	> 7200*	-
200	LT	2854	631	4720	515	> 7200*	**(272447, 284455; 4.2%)**	-	> 7200*	-
200	TT	3042	698	4664	564	> 7200*	**(307127, 324974; 5.4%)**	-	> 7200*	-
					Geometric Mean:	1			4.91	

-: time limit of 7200 seconds reached at pre-solve stage.
*: 7200 was used when computing the geometric mean.

that were solved in less than the computational time that is given by the x-axis of the plot. As can be seen from Figure 3, the cutting plane approach outperforms GUROBI for all the problem variants. Furthermore the performance profiles clearly shows that CSAHLP is the most challenging to solve for both, the proposed cutting plane approach and GUROBI.

5.4. Increasing the computational time limit

As discussed earlier, CSHALP is more computationally challenging than SAHLP and SApHMP and a relatively large optimality gap (up to 79%) remained for some of the instances that are

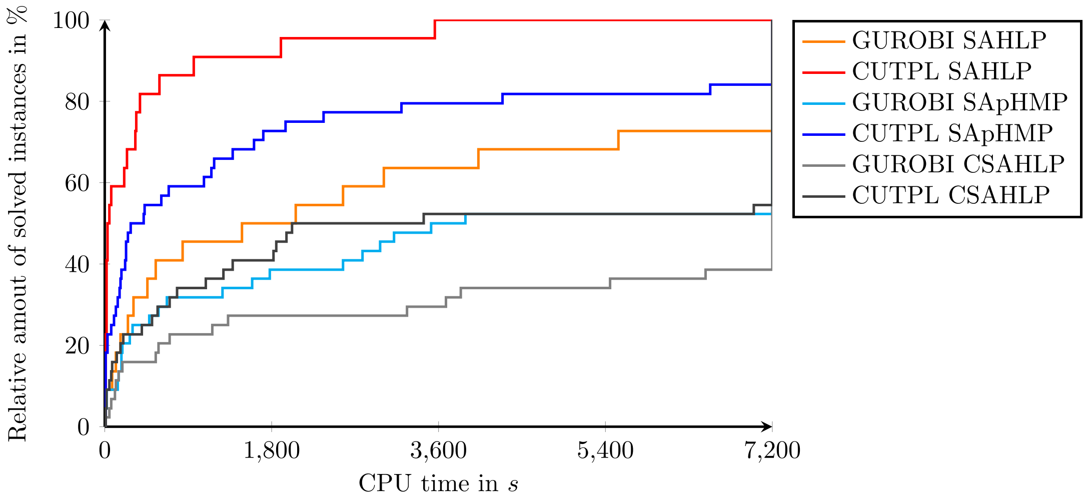


Figure 3 **Performance profiles for SAHLP, SApHMP, CSAHLP**

presented in Table 3. For those instances that were not solved to optimality using the proposed cutting plane approach, the imposed computational time limit is increased to 4 hours and the corresponding results are summarized in Table 4. Increasing the computational time leads to the solution of only 2 of the 20 previously unsolved instances. Overall, the average gap decreased from 11.9% to 6.6%. While increasing the computational time limit beyond 4 hours might potentially lead to the solution of the remaining instances, future research investigating a customized approach for the particular case of CSHALP can potentially be more promising in solving larger instances.

Table 4 Computational results with 4 hours time limit for CSAHLP

Instance		Cutting Plane (2 hours time limit)			Cutting Plane (4 hours time limit)		
N	Type	CUTS	Nodes	Opt.	CUTS	Nodes	Opt.
40	TT	1519	219441	(379266, 379805; 0.1%)	1519	404238	(379657, 379783; 0.0%)
60	LT	2437	85420	(273282, 274154; 0.3%)	2439	269080	(273535, 274075; 0.2%)
75	LL	2301	575	(276436, 327569; 15.6%)	2802	86069	284389
75	LT	2808	14092	(302231, 309403; 2.3%)	2905	88886	(303257, 303479; 0.1%)
75	TT	1726	160886	(428650, 429247; 0.1%)	1728	297994	(428991, 429205; 0.0%)
90	LT	2779	2289	(281331, 306430; 8.2%)	3799	74295	(288402, 289156; 0.3%)
90	TT	2301	4630	(422624, 544419; 22.3%)	2893	41525	(428066, 428432; 0.1%)
100	LT	2470	560	(270716, 303620; 10.9%)	2946	105759	279161
100	TT	2249	5106	(510734,686823; 25.6%)	2562	24373	(511935, 591384; 13.4%)
125	LT	3173	540	(260806, 271381; 3.9%)	5308	10403	(267536, 269486; 0.7%)
150	LL	4317	547	(228864, 232100; 1.4%)	4317	578	(228864, 232100; 1.4%)
150	LT	3294	541	(238671, 269319; 11.4%)	3294	551	(238692, 269319; 11.4%)
150	TT	3408	685	(291676, 376356; 22.5%)	3939	14090	(301355, 303188; 0.6%)
175	LL	6390	540	(236107, 238449; 1.0%)	6390	572	(236107, 238449; 1.0%)
175	LT	4689	561	(249586, 254953; 2.1%)	4689	587	(249616, 254953; 2.1)%
175	TT	3030	1047	(287615, 318347; 9.7%)	3636	131390	(294231, 294528; 0.1%)
200	LL	4669	516	(253512, 287265; 11.7%)	4669	520	(253545, 287265; 11.7%)
200	TL	4353	519	(262228, 1247994; 79.0%)	4353	526	(262558, 1247994; 79.0%)
200	LT	4720	515	(272447, 284455; 4.2%)	4720	520	(272569, 284455; 4.2%)
200	TT	4664	564	(307127, 324974; 5.4%)	4664	607	(307233, 324974; 5.5%)
		Minimum Gap:		0.1%			0%
		Average Gap:		11.9%			6.6%
		Maximum Gap:		79.0%			79.0%

Table 5 Effect of Number of Scenarios on the Cutting Plane Approach (SAHLP).

N	Type	#Scenarios	CUTS0	CPU0	CUTS	Nodes	CPU
25	L	5	197	0	239	332	1
25	L	10	627	1	719	521	5
25	L	15	1123	2	1624	36	3
25	L	20	291	0	291	0	1
25	L	25	0	0	0	0	1
25	T	5	125	0	125	0	0
25	T	10	0	0	0	0	0
25	T	15	0	0	0	0	0
25	T	20	0	0	0	0	0
25	T	25	0	0	0	0	0
50	L	5	802	3	1105	521	20
50	L	10	1338	5	1926	646	102
50	L	15	1922	8	2363	9371	78
50	L	20	2726	14	5059	1947	618
50	L	25	3351	14	4089	12621	1231
50	T	5	235	1	235	0	1
50	T	10	1157	6	1889	576	36
50	T	15	1807	10	3130	13115	499
50	T	20	2446	13	3466	525	138
50	T	25	1788	18	4350	170	38

5.5. Effect of Additional Scenarios

The results of the prior sections included 5 scenarios. While this choice of the number of scenarios is consistent with the literature such as Alumur et al. (2012) where also 5 scenarios are considered, it is expected that the capabilities of solving the instances to optimality is strongly dependent on the number of scenarios that are considered. Thus this section evaluates the impact of increasing the number of scenarios on the computational performance of the proposed cutting plane approach.

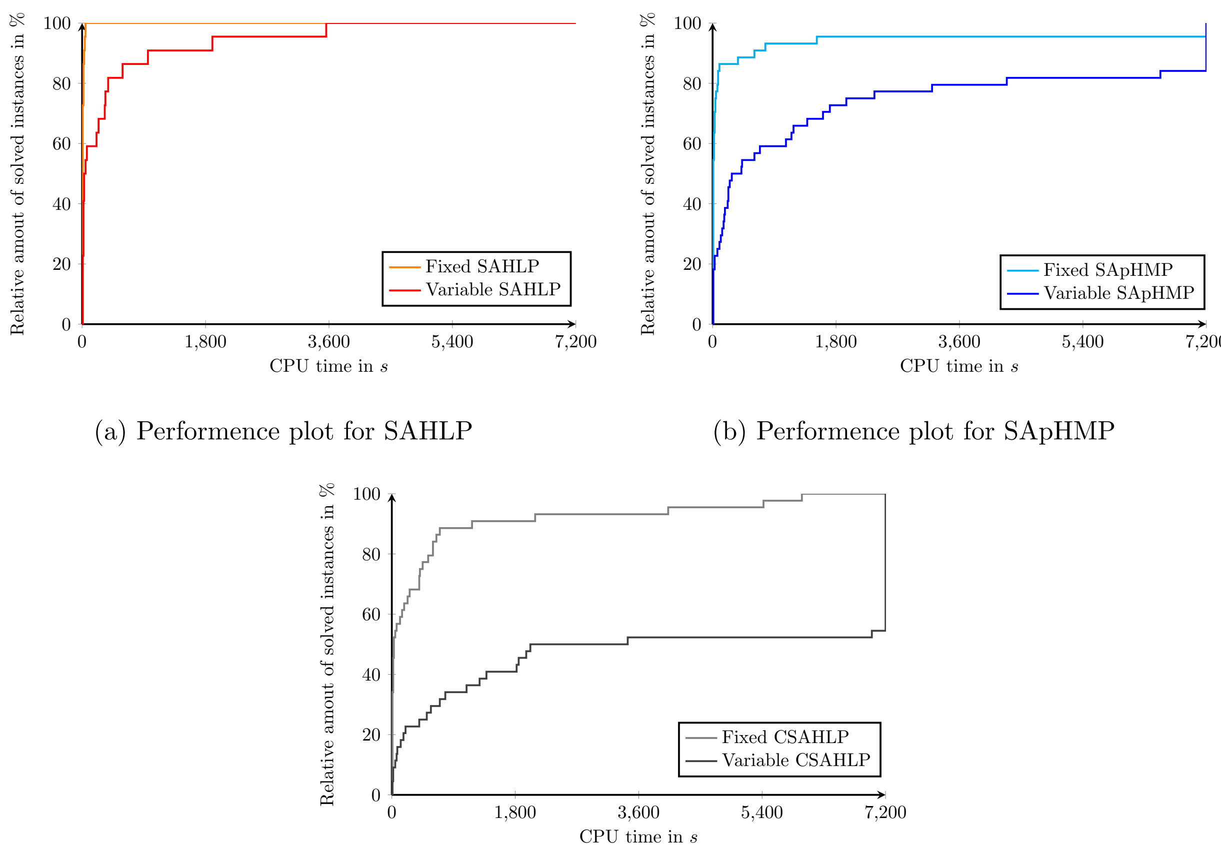
Since, as expected the computational time increases relatively quickly as the number of scenarios increases, only results for the instances with 25 and 50 nodes are reported since the majority of these instances can be solved to optimality within the time limit as the number of scenarios is varied from 5 to 25. The results for SAHLP, CSAHLP, and SApHMP which are summarized in Tables 5–7, respectively, show that the increase in the number of scenarios consistently leads to an increase in the number of cuts that are generated at the root node as well as the full branch-and-cut tree. The general trend for the overall computational time for SAHLP, SApHMP, and CSAHLP is also increasing as the number of scenarios increases. CSAHLP remains the most challenging to solve where instances with 50 nodes and 10 or more scenarios fail to solve within the time limit. While for SAHLP and SApHMP all the instances with 25 and 50 nodes are solved to optimality, the quick increase in computational time is noticeable particularly for SApHMP with $p = 5$ where the computational time increased from 6 seconds to 1141 seconds for the case with 50 nodes as the number of scenarios increased from 5 to 25.

Table 6 Effect of Number of Scenarios on the Cutting Plane Approach (SApHMP).

N	Type	#Scenarios	CUTS0	CPU0	CUTS	Nodes	CPU
25	2	5	197	0	234	209	0
25	2	10	630	1	717	1024	4
25	2	15	861	1	1088	1180	105
25	2	20	1312	21	2770	1110	39
25	2	25	1546	61	4157	677	361
25	3	5	382	0	841	190	3
25	3	10	781	2	1510	433	10
25	3	15	955	3	2542	575	14
25	3	20	1436	17	2512	630	30
25	3	25	1807	19	2506	757	40
25	4	5	476	0	1210	1077	556
25	4	10	926	1	2128	1592	695
25	4	15	1377	4	2966	1091	2113
25	4	20	1656	16	4066	2775	2167
25	4	25	2021	10	4673	11092	2613
25	5	5	464	0	1230	2037	195
25	5	10	824	1	2175	33226	753
25	5	15	1219	4	2908	25787	1936
25	5	20	1562	8	4197	18971	2225
25	5	25	1553	8	4427	44105	1759
50	2	5	291	2	373	519	6
50	2	10	594	5	1110	536	19
50	2	15	1979	10	2337	542	204
50	2	20	2639	16	3005	592	416
50	2	25	1346	17	1908	543	60
50	3	5	468	3	628	396	8
50	3	10	885	6	1410	727	337
50	3	15	1730	10	2548	1785	240
50	3	20	3015	14	3699	3282	597
50	3	25	2933	21	4814	747	933
50	4	5	696	2	927	174	5
50	4	10	1532	5	2007	564	462
50	4	15	2337	9	2918	882	200
50	4	20	3346	15	3852	2951	980
50	4	25	3289	16	5084	3895	957
50	5	5	675	2	839	260	6
50	5	10	1296	6	1949	610	120
50	5	15	1966	9	2929	684	343
50	5	20	3251	14	3945	7644	689
50	5	25	3168	14	4717	2139	1141

Table 7 Effect of Number of Scenarios on the Cutting Plane Approach (CSAHLP).

N	Type	#Scenarios	CUTS0	CPU0	CUTS	Nodes	CPU
25	LL	5	359	0	509	256	1
25	LL	10	633	0	744	519	3
25	LL	15	904	1	1433	519	7
25	LL	20	824	4	1406	121	6
25	LL	25	1040	5	1554	545	520
25	LT	5	344	0	523	1267	3
25	LT	10	534	1	638	231	2
25	LT	15	606	1	709	526	5
25	LT	20	570	2	678	532	10
25	LT	25	637	3	756	521	9
25	TL	5	317	0	412	1169	2
25	TL	10	788	1	1133	1174	265
25	TL	15	860	1	1913	847	436
25	TL	20	1119	2	2032	903	837
25	TL	25	1892	5	2741	572	682
25	TT	5	410	0	573	459	1
25	TT	10	528	1	655	531	9
25	TT	15	945	0	1152	717	26
25	TT	20	986	3	1221	530	65
25	TT	25	1026	3	1273	538	79
50	LL	5	892	4	1071	536	22
50	LL	10	1458	6	1896	841	473
50	LL	15	2072	15	3398	3530	1282
50	LL	20	3206	27	6830	18653	>7200
50	LL	25	4108	26	6105	113754	>7200
50	LT	5	726	4	1765	8403	1840
50	LT	10	1793	12	3075	55517	>7200
50	LT	15	2573	22	4937	87266	>7200
50	LT	20	3373	34	6350	45167	>7200
50	LT	25	3460	60	5175	764	>7200
50	TL	5	538	4	921	1286	60
50	TL	10	1524	9	3050	928	92
50	TL	15	1262	17	3054	6747	357
50	TL	20	3143	35	5589	570	>7200
50	TL	25	3495	47	7969	723	>7200
50	TT	5	647	5	958	331900	6216
50	TT	10	1419	14	2989	141965	>7200
50	TT	15	1965	27	4587	69687	>7200
50	TT	20	2983	39	6147	21270	>7200
50	TT	25	3488	64	5303	7923	>7200



(a) Performence plot for SAHLP

(b) Performence plot for SApHMP

(c) Performence plot for CSAHLP

Figure 4 Variable allocation vs fixed allocation performance profiles.

5.6. Variable Allocation vs Fixed Allocation

The variable allocation problem that is the main focus of this paper is computationally more challenging to solve than the fixed allocation problem. To illustrate this, the computational time for solving SAHLP, SApHMP, and CSAHLP with variable allocation is compared to solving the same problems with fixed allocation. To solve the fixed allocation variation, the cutting plane approach that is proposed in Section 4 is modified by using the expected demand value (following Theorem 1) and replacing x^s_{ik} in MP_v by x_{ik}, thus forcing all the allocations to be the same in all the scenarios. The same instances that are detailed in Tables 1–3 are solved with fixed allocation and the performance profiles that are shown in Figure 4 show the number of instances that were solved in less than the computational time that is given by the x-axis of the plot. The performance profiles clearly show that for SAHLP, SApHMP, and CSAHLP the variable allocation is significantly more challenging than fixed allocation. For SAHLP, Figure 4a shows that all the instances for the fixed allocation are solved in less than 50 seconds while the majority of the instances for the variable allocation consume more time. The same result is observed in Figure 4b for SApHMP where the majority of the instances are solved in less than 1000 seconds of computational time for the fixed allocation case while the the instances for the variable allocation case are more challenging. Finally as shown in Figure 4c, CSAHLP with variable allocation is the most challenging while for the fixed allocation all the instances are solved to optimality.

6. Conclusion

This paper presented the single allocation hub location problem with variable allocation and proposed a mixed-integer nonlinear programming formulation and a customized solution approach based on cutting planes. As shown in the paper the variable allocation problem is significantly more challenging to solve that the fixed allocation problem as the latter can be solved by only considering the expected values of the random variables. The proposed cutting plane approach is implemented using a branch-and-cut framework where the cuts are efficiently separated without the need to solve a subproblem using an optimization solver. While only demand uncertainty was considered in this paper, the proposed approach is independent of the demand uncertainty and can be applied to the cases that include uncertainty in the allocation costs and flow routing costs. Extensive computational tests on instances from the literature highlighted the advantages of the proposed approach compared to L-shaped decomposition and the direct solution of the deterministic equivalent formulation using GUROBI.

Acknowledgments

Nicolas Kämmerling has been supported by the German Research Foundation (DFG) under grant CL 318/14. Joe Naoum-Sawaya was supported by NSERC Discovery Grant RGPIN-2017-03962.

References

Alumur, S. A. and Kara, B. Y. (2008). Network hub location problems: The state of the art. *European Journal of Operational Research*, 190(1):1–21.

Alumur, S. A., Nickel, S., and Saldanha-da Gama, F. (2012). Hub location under uncertainty. *Transportation Research Part B: Methodological*, 46(4):529–543.

Beasley, J. E. (2012). OR library. `http://people.brunel.ac.uk/~mastjjb/jeb/orlib/phubinfo.html`.

Campbell, J. F. (1996). Hub location and the p-hub median problem. *Operations Research*, 44(6):923–935.

Campbell, J. F., Ernst, A. T., and Krishnamoorthy, M. (2005a). Hub arc location problems: part iiformulations and optimal algorithms. *Management Science*, 51(10):1556–1571.

Campbell, J. F., Ernst, A. T., and Krishnamoorthy, M. (2005b). Hub arc location problems: part iintroduction and results. *Management Science*, 51(10):1540–1555.

Campbell, J. F. and O'Kelly, M. E. (2012). Twenty-five years of hub location research. *Transportation Science*, 46(2):153–169.

Carlsson, J. G. and Jia, F. (2013). Euclidean hub-and-spoke networks. *Operations Research*, 61(6):1360–1382.

Contreras, I., Cordeau, J.-F., and Laporte, G. (2011a). Benders decomposition for large-scale uncapacitated hub location. *Operations Research*, 59(6):1477–1490.

Contreras, I., Cordeau, J.-F., and Laporte, G. (2011b). Stochastic uncapacitated hub location. *European Journal of Operational Research*, 212(3):518–528.

Contreras, I., Díaz, J. A., and Fernández, E. (2011c). Branch and price for large-scale capacitated hub location problems with single assignment. *INFORMS Journal on Computing*, 23(1):41–55.

Correia, I., Nickel, S., and Saldanha-da Gama, F. (2010). Single-assignment hub location problems with multiple capacity levels. *Transportation Research Part B: Methodological*, 44(8):1047–1066.

Dolan, E. D. and Moré, J. J. (2002). Benchmarking optimization software with performance profiles. *Mathematical programming*, 91(2):201–213.

Duran, M. A. and Grossmann, I. E. (1986). An outer-approximation algorithm for a class of mixed-integer nonlinear programs. *Mathematical Programming*, 36(3):307–339.

Elhedhli, S. and Wu, H. (2010). A lagrangean heuristic for hub-and-spoke system design with capacity selection and congestion. *INFORMS Journal on Computing*, 22(2):282–296.

Ernst, A. T. and Krishnamoorthy, M. (1996). Efficient algorithms for the uncapacitated single allocation p-hub median problem. *Location science*, 4(3):139–154.

Ernst, A. T. and Krishnamoorthy, M. (1998). Exact and heuristic algorithms for the uncapacitated multiple allocation p-hub median problem. *European Journal of Operational Research*, 104(1):100–112.

Ernst, A. T. and Krishnamoorthy, M. (1999). Solution algorithms for the capacitated single allocation hub location problem. *Annals of Operations Research*, 86:141–159.

Harremoës, P. (2001). Binomial and poisson distributions as maximum entropy distributions. *IEEE Transactions on information theory*, 47(5):2039–2041.

Jaillet, P., Song, G., and Yu, G. (1996). Airline network design and hub location problems. *Location science*, 4(3):195–212.

Klincewicz, J. G. (1998). Hub location in backbone/tributary network design: a review. *Location Science*, 6(1):307–335.

Labbé, M. and Yaman, H. (2004). Projecting the flow variables for hub location problems. *Networks*, 44(2):84–93.

Laporte, G. and Louveaux, F. V. (1993). The integer l-shaped method for stochastic integer programs with complete recourse. *Operations research letters*, 13(3):133–142.

Marianov, V. and Serra, D. (2003). Location models for airline hubs behaving as M/D/c queues. *Computers & Operations Research*, 30(7):983–1003.

Meier, J. F. and Clausen, U. (2017). Solving single allocation hub location problems on euclidean data. *Transportation Science*.

Meier, J. F., Clausen, U., Rostami, B., and Buchheim, C. (2016). A compact linearisation of euclidean single allocation hub location problems. *Electronic Notes in Discrete Mathematics*, 52:37–44.

Nickel, S., Schöbel, A., and Sonneborn, T. (2001). Hub location problems in urban traffic networks. In *Mathematical methods on optimization in transportation systems*, pages 95–107. Springer.

O'Kelly, M. E. (1987). A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32(3):393–404.

Qin, Z. and Gao, Y. (2017). Uncapacitated p-hub location problem with fixed costs and uncertain flows. *Journal of Intelligent Manufacturing*, 28(3):705–716.

Rostami, B., Buchheim, C., Meier, J. F., and Clausen, U. (2016). Lower bounding procedures for the single allocation hub location problem. *Electronic Notes in Discrete Mathematics*, 52:69–76.

Rostami, B., Kämmerling, N., Buchheim, C., and Clausen, U. (2018). Reliable single allocation hub location problem under hub breakdowns. *Computers & Operations Research*, 96:15–29.

Skorin-Kapov, D., Skorin-Kapov, J., and O'Kelly, M. (1996). Tight linear programming relaxations of uncapacitated p-hub median problems. *European Journal of Operational Research*, 94(3):582–593.

Tran, T. H., OHanley, J. R., and Scaparra, M. P. (2016). Reliable hub network design: Formulation and solution techniques. *Transportation Science*, 51(1):358–375.

Yang, T.-H. (2009). Stochastic air freight hub location and flight routes planning. *Applied Mathematical Modelling*, 33(12):4424–4430.

Appendix A: L-Shaped Decomposition

The L-shaped decomposition is a common approach to solve two-stage stochastic programs with recourse with a finite number of realizations. In this Appendix, we illustrate the application of the L-shaped decomposition to SAHLP. As detailed next, the L-shaped decomposition takes advantage of the decomposable structure of the problem however the overall computational performance fails to compete with the cutting plane approach that was proposed in Section 4.

A.1. Decomposing the Problem

Applying the L-shaped decomposition to $\text{DEF}_{\mathtt{v}}$ decomposes the problem into an integer master problem and a binary quadratic subproblem. The hub location decisions, i.e. the z variables, are incorporated in the master problem, and then for a given choice of location variables $\hat{z}$, m subproblems are solved to obtain the corresponding optimal values of the second stage allocation variables. Particularly, by projecting $\text{DEF}_{\mathtt{v}}$ on the space defined by the z variables, the following master problem is obtained:

$$\begin{aligned} \text{MP}_{\mathtt{v}}: \quad & \min \quad \sum_{k\in N} f_k z_k + \sum_{s\in S_w} p_s \eta_s \\ & \text{s.t.} \quad (19), (20) \\ & \qquad \eta_s \geq \phi_s(z), \end{aligned}$$

where $\phi_s(z)$ is the minimum allocation and routing costs for the current locations z under scenario s. Note that constraints (19) and (20) are enough to ensure feasibility and $\phi_s(z)$ is bounded.

A natural way to solve $\text{MP}_{\mathtt{v}}$ is to find an affine function $\pi_s^T z + \pi_s^0$ that underestimates $\phi_s(z)$, i.e. $\pi_s^T z + \pi_s^0 \leq \phi_s(z)$, on the feasible region such that the estimate is tight at the optimal solution z^*. The integer L-shaped cuts (Laporte and Louveaux 1993) provide such an affine function. Let H be the index set of the chosen hubs at the first stage and L_s a known lower bound on $\phi_s(z)$, then the following integer optimality cut at $\hat{z}$ is obtained

$$\eta_s \geq (\bar{\phi}_s(\hat{z}) - L_s)\Big(\sum_{i\in H} z_i - \sum_{i\notin H} z_i + 1 - |H|\Big) + L_s \quad s \in S_w. \tag{44}$$

Given a feasible solution $(\hat{z}, \hat{\eta}_s)$ of $\text{MP}_{\mathtt{v}}$, the corresponding integer L-shaped cuts are added to the master problem. Iterating the procedure provides an exact solution method in the spirit of Benders' decomposition. However, given that at each iteration of the algorithm, m binary quadratic programs (or their corresponds MILPs) are solved, the integer L-shaped cuts can be complemented with other inequalities. One way to construct such inequalities is to use the subgradient cuts that are given by the continuous relaxation of $\phi_s(z)$. For each $s \in S_w$, let $\bar{\phi}_s(z)$ represent the continuous relaxation of $\phi_s(z)$ and $u_s \in \partial\bar{\phi}_s(\hat{z})$ be a subgradient of $\bar{\phi}_(s)$ at $\hat{z}$, then the subgradient cut is given by

$$\eta_s \geq \bar{\phi}_s(\hat{z}) + u_s(z - \hat{z}) \quad s \in S_w. \tag{45}$$

The L-shaped decomposition is implemented in a branch-and-cut framework similar to the cutting plane approach of Section 4 such that the violated cuts (44) and (45) are added at the root node and at nodes where a candidate incumbent is available. If no violated cut exist, then the algorithm proceeds by branching on one of the binary variables z that has a non-binary value at the current node of the tree. An optimal solution is reached when all the nodes in the branching tree are fathomed.

A.2. Solving the Subproblem

At each node of the Branch-and-Bound tree, an optimal solution $\hat{z}$ at that node of the tree is obtained. To solve, the corresponding subproblem which is a binary quadratic program, we adapt the flow-based linearization technique of Ernst and Krishnamoorthy (1996) to obtain an equivalent MILP reformulation. Thus for a given $s \in S_w$, the following primal subproblem PS$(\hat{z}, s)$ is solved

$$\phi_s(\hat{z}) = \min \sum_{\substack{i,k \in N \\ i \neq k}} \bar{c}^s_{ik}\, x^s_{ik} + \sum_{\substack{i,k \in N \\ i \neq k}} \sum_{\ell \in N} \alpha\, d_{k\ell} y^s_{ik\ell}$$

$$\text{s.t.} \quad \sum_{k \in N: k \neq i} x^s_{ik} = 1 - \hat{z}_i \quad \forall i \in N \tag{46}$$

$$x^s_{ik} \leq \hat{z}_k \quad \forall i,k \in N, i \neq k \tag{47}$$

$$\sum_{\ell \in N} y^s_{ik\ell} - \sum_{\substack{\ell \in N \\ \ell \neq i}} y^s_{i\ell k} = O^s_i x^s_{ik} - \sum_{j \in N: j \neq k} w^s_{ij} x^s_{jk} - w^s_{ik} \hat{z}_k \quad \forall i,k,\ i \neq k \tag{48}$$

$$\sum_{\ell \in N} y^s_{ik\ell} \leq O^s_i x^s_{ik} \quad \forall i,k,\ i \neq k \tag{49}$$

$$x^s_{ik} \in \{0,1\} \quad \forall i,k \in N,\ i \neq k$$

$$y^s_{ik\ell} \geq 0 \quad \forall i,k,\ell \in N.$$

where for each $i,k \in N$, $i \neq k$ and $s \in S_w$

$$\bar{c}^s_{ik} = c^s_{ik} + \sum_{\ell \in N} \alpha\, w^s_{i\ell} d_{k\ell} \hat{z}_\ell.$$

Note that because of constraints (19), PSub$(\hat{z}, s)$ is always feasible and integer optimality cuts (44) are generated by solving PS$(\hat{z}, s)$ for each $s \in S_w$. Furthermore, to obtain the subgradient cuts (45), consider the LP relaxation of PS$(\hat{z}, s)$ and let $(\hat{x}, \hat{y})$ be the optimal solution. Moreover, let $\hat{\beta}^s_i$, $\hat{\lambda}^s_{ik}$, $\hat{\mu}^s_{ik}$, and $\hat{v}^s_{ik}$ be the optimal dual variables associated with constraints (46), (47), (48), and (49), respectively. The subgradiant cut at $(\hat{x}, \hat{y})$ is then

$$\eta_s \geq \sum_{k \in N} \hat{\beta}^s_k (1 - z_k) + \sum_{\substack{i,k \in N \\ i \neq k}} (\hat{\lambda}^s_{ik} - w^s_{ik} \hat{\mu}^s_{ik}) z_k. \tag{50}$$

A.3. Computational Evaluation

The L-shaped decomposition is implemented in C++ using GUROBI 6.5 callback framework. The results for SAHLP and SApHMP are shown in Tables 8 and 9, respectively. These results are based on the same dataset with the same 5 scenarios as the ones used in Section 5. The results show that the cutting plane approach consistently outperforms the L-shaped decomposition which also performs poorly compared to Gurobi. Several SHALP and SApHMP were not solved to optimality within the 2 hours time limit using the L-shaped method though an advantage over Gurobi is that upper and lower bounds are returned for all the instances. We note that CSAHLP was not evaluated due to the fact that the L-shaped method performs poorly on SAHLP and SApHMP which are relatively less computationally challenging while CSAHLP is more challenging and the application of the L-shaped approach requires the generation of feasibility cuts to eliminate capacity violations.

Table 8 L-Shaped Decomposition Computational Results for SAHLP.

Instance		GUROBI			L-Shaped Decomposition				Cutting Plane			
N	type	Nodes	CPU	Opt.	CUTS	Nodes	CPU	Opt.	CUTS	Nodes	CPU	Opt.
25	L	0	8	203271	194	369	151	203271	239	332	1	203271
25	T	0	4	240689	39	57	40	240689	125	0	0	240689
40	L	5	250	257241	890	3001	2099	257241	1153	171	5	257241
40	T	0	74	307257	65	82	149	307257	930	56	3	307257
50	L	0	161	219345	425	1316	1766	219345	1105	521	20	219345
50	T	0	120	277276	115	162	548	277276	235	0	1	277276
60	L	0	307	230305	600	1235	4496	230305	890	521	17	230305
60	T	12	459	275062	95	89	963	275062	1155	1664	26	275062
75	L	40	2059	271648	450	1247	>7200	(244221, 272031; 10.2%)	2258	2418	236	271648
75	T	0	549	325953	135	265	1299	325953	1259	1034	21	325953
90	L	3	4029	244711	515	1120	>7200	(215722, 257351; 16.2%)	2049	714	339	244711
90	T	0	840	278762	165	207	4629	278762	1400	3	16	278762
100	L	0	2563	251871	310	871	>7200	(219295, 262362; 16.4%)	1781	1439	210	251871
100	T	0	1472	331972	65	102	2211	331972	794	5	14	331972
125	L	-	>7200	-	225	510	>7200	(189134, 243116; 22.2%)	3300	3863	376	233366
125	T	0	3002	262691	125	222	6934	262691	795	522	47	262691
150	L	-	>7200	-	190	510	>7200	(112285, 242871; 53.8%)	2137	1584	589	241412
150	T	0	5532	256235	60	195	6242	256235	960	520	64	256235
175	L	-	>7200	-	105	512	>7200	(192322, 260712; 26.2%)	3117	2937	953	232238
175	T	-	>7200	-	300	510	>7200	(240639, 257698; 6.6%)	2016	780	323	251119
200	L	-	>7200	-	120	510	>7200	(97528, 281991; 65.4%)	4690	1484	3555	251050
200	T	-	>7200	-	300	510	>7200	(225067, 270180; 16.7%)	3891	3828	1892	260868

-: time limit of 7200 seconds reached at pre-solve stage.

Table 9 L-Shaped Decomposition Computational Results for SApHMP.

Instance		GUROBI			L-Shaped Decomposition				Cutting Plane			
N	p	Nodes	CPU	Opt.	CUTS	Nodes	CPU	Opt.	CUTS	Nodes	CPU	Opt.
25	2	0	7	146259	100	59	35	146259	509	256	1	146259
25	3	0	8	125011	264	399	133	125011	412	1169	2	125011
25	4	0	16	110716	685	1565	411	110716	523	1267	2	110716
25	5	0	11	97740	865	1903	582	97740	573	459	1	97740
40	2	30	146	197565	215	198	426	197565	796	830	22	197565
40	3	22	263	175623	1060	1992	1757	175623	1361	1821	66	175623
40	4	128	475	159074	3160	8777	5328	159074	1453	1254	91	159074
40	5	323	571	144945	3450	7906	>7200	(127793, 146234; 12.6%)	1279	26605	247	144945
50	2	0	132	157233	200	120	890	157233	373	519	5	157233
50	3	0	185	139851	840	1647	3621	139851	628	396	8	139851
50	4	0	176	124972	1800	5454	6217	124972	927	174	5	124972
50	5	0	171	113818	2120	7440	>7200	(104829, 114102,8.1%)	839	260	6	113818
60	2	0	295	181998	245	234	1349	181998	920	558	23	181998
60	3	0	663	162309	1240	2846	6272	162309	1467	1170	230	162309
60	4	11	1771	148832	1385	1978	>7200	(121332, 151525; 19.9%)	1671	5747	211	148832
60	5	50	2772	137135	1510	1835	>7200	(104933, 140252; 25.2%)	1981	2284	1605	137135
75	2	3	1587	217498	350	333	3417	217498	1157	732	156	217498
75	3	3	2566	192734	735	1045	>7200	(168325, 194637; 13.5%)	2150	1112	166	192734
75	4	23	2969	175058	940	1849	>7200	(145661, 179170; 18.7%)	1905	1608	180	175058
75	5	0	1270	159837	800	1606	>7200	(124026, 166532; 25.5%)	1663	1325	120	159837
90	2	45	3513	200327	473	422	>7200	(189351, 200327; 5.4%)	1634	1003	601	200327
90	3	-	>7200	-	515	1051	>7200	(145840, 180152; 19.0%)	2201	2450	278	176290
90	4	47	3889	158724	430	630	>7200	(120267, 161017; 25.3%)	2017	1374	426	158724
90	5	-	>7200	-	415	510	>7200	(74774.6, 154743; 51.6%)	3196	3417	1174	149298
100	2	3	3120	191461	355	649	>7200	(183066, 191559; 4.4%)	1052	940	140	191461
100	3	-	>7200	-	375	837	>7200	(134599, 177946; 24.3%)	3649	562	>7200	(170047, 173101; 1.8%)
100	4	-	>7200	-	440	456	>7200	(101787, 167964; 39.4%)	2307	4849	413	154798
100	5	-	>7200	-	490	695	>7200	(95597.5, 149045; 35.8%)	3320	9513	1379	144208
125	2	-	>7200	-	360	510	>7200	(164729, 186739; 11.7%)	2796	2649	683	186248
125	3	-	>7200	-	300	510	>7200	(127954, 170852; 25.1%)	4811	5275	1709	166765
125	4	-	>7200	-	275	510	>7200	(66601.3, 162452; 59.0%)	4540	6061	2352	153478
125	5	-	>7200	-	480	259	>7200	(57926.7, 147219; 60.6%)	4981	10349	6530	143680
150	2	-	>7200	-	255	510	>7200	(151966, 184545; 17.6%)	2347	979	221	196615
150	3	-	>7200	-	195	510	>7200	(112085, 177659; 36.9%)	3920	2068	1147	175416
150	4	-	>7200	-	215	510	>7200	(98756.5, 152699; 35.3%)	2869	11768	1068	158099
150	5	-	>7200	-	205	510	>7200	(67209.3, 144500; 53.4%)	3210	55183	1942	144802
175	2	-	>7200	-	325	510	>7200	(158175, 186457; 15.2%)	2086	19584	3195	186457
175	3	-	>7200	-	155	510	>7200	(99211.3, 175000; 43.3%)	4259	522	>7200	(165675, 168716; 1.8%)
175	4	-	>7200	-	155	510	>7200	(101425, 155346; 34.7%)	4474	525	>7200	(150052, 154176; 2.7%)
175	5	-	>7200	-	220	510	>7200	(55762.7, 162151; 65.6%)	4921	522	>7200	(139550, 144827; 3.6%)
200	2	-	>7200	-	270	510	>7200	(169583, 205604; 17.5%)	2893	947	4283	203691
200	3	-	>7200	-	180	510	>7200	(128319, 182225; 29.6%)	3577	701	>7200	(178223, 178832; 0.3%)
200	4	-	>7200	-	240	510	>7200	(108465, 170005; 36.2%)	5599	519	>7200	(162189, 165820; 2.2%)
200	5	-	>7200	-	115	510	>7200	(68387, 166973; 59.0%)	5492	518	>7200	(151382, 156660; 3.4%)

-: time limit of 7200 seconds reached at pre-solve stage.