**Submitted to *Operations Research***
**manuscript OPRE-2019-07-345.R1**

# Single Allocation Hub Location with Heterogeneous Economies of Scale

Borzou Rostami
Lazaridis School of Business and Economics, Wilfrid Laurier University and
CIRRELT and CERC-Data Science for Real-time Decision-making, Polytechnique Montréal, Canada
brostami@wlu.ca

Masoud Chitsaz
Kinaxis and HEC Montréal, Canada
masoud.chitsaz@hec.ca

Okan Arslan
HEC Montréal and CIRRELT, Canada
okan.arslan@hec.ca

Gilbert Laporte
HEC Montréal, Canada and School of Management, University of Bath, United Kingdom
gilbert.laporte@cirrelt.ca

Andrea Lodi
CERC-Data Science for Real-time Decision-making, Polytechnique Montréal and GERAD, Canada
andrea.lodi@polymtl.ca

We study the single allocation hub location problem with heterogeneous economies of scale (SAHLP-h). The SAHLP-h is a generalization of the classical single allocation hub location problem (SAHLP), in which the hub-hub connection costs are piecewise linear functions of the amounts of flow. We model the problem as an integer non-linear program, which we then reformulate as a mixed integer linear program (MILP) and also as a mixed integer quadratically constrained program (MIQCP). We exploit the special structures of these models to develop Benders type decomposition methods with integer subproblems. We use an integer L-shaped decomposition to solve the MILP formulation. For the MIQCP, we dualize a set of complicating constraints to generate a Lagrangian function, which offers us a subproblem decomposition and a tight lower bound. We develop linear dual functions to underestimate the integer subproblem, which helps us obtain optimality cuts with a convergence guarantee by solving a linear program. Moreover, we develop a specialized polynomial-time algorithm to generate enhanced cuts. To evaluate the efficiency of our models and solution approaches, we perform extensive computational experiments on both uncapacitated and capacitated SAHLP-h instances derived from the classical Australian Post dataset. The results confirm the efficacy of our solution methods in solving large-scale instances.

*Key words*: Single allocation, hub location, economies of scale, quadratic program, Benders decomposition, Lagrangian relaxation

## 1. Introduction

The purpose of this paper is to study the single allocation hub location problem (SAHLP) with heterogeneous economies of scale. Economies of scale in the hub location literature have traditionally been modeled by multiplying the hub-hub connection costs by a fixed parameter $\alpha \in (0, 1]$ (Alumur and Kara 2008, Campbell and O'Kelly 2012, Contreras 2015). Even though the economies are due to the amalgamation of flows, this fact is not taken into account when the discount factor is fixed. In our study, we develop a SAHLP formulation with a piecewise linear cost function for each arc, hence the name *heterogeneous economies of scale*. Based on the structure of this formulation, we also develop advanced decomposition methods with efficiently solvable separation subproblems.

The hub-and-spoke network structure is pervasive in several important applications including telecommunication systems (Klincewicz 1998, Carello et al. 2004), airline services (Jaillet et al. 1996, Eiselt and Marianov 2009, Dukkanci and Kara 2017), liner shipping (Imai et al. 2009, Gelareh and Pisinger 2011), postal delivery services (Ernst and Krishnamoorthy 1996, Çetiner et al. 2010), and public transportation (Nickel et al. 2001, Mahéo et al. 2019). Given a hub-and-spoke network with origin-destination commodity flows, a node is either a hub or is assigned to a hub to route its commodities. Hubs are used to sort, consolidate, and redistribute flows. This way, the flows are routed in a network of hubs and are shipped to their destinations. The main purpose of opening hubs is to benefit from the economies of scale. Whereas opening and operating hubs imply extra costs, the increased amounts of flows between hubs decrease the unit costs. Therefore the location of hubs is an important factor in cost minimization. The hub location problem minimizes the total cost of hub setup and flow transportation by selecting the hub nodes, and allocating the origin and destination nodes to the selected hubs.

Hub location problems can be classified as either single allocation hub location problems (O'Kelly 1987, Ernst and Krishnamoorthy 1996) or multiple allocation problems (Campbell 1996, Ernst and Krishnamoorthy 1998, Contreras et al. 2011a, 2012). In the first case, all flows to and from a spoke are forced onto a single access arc, while in the multiple allocation case, the flows to and from a spoke may be divided among the incident access arcs. In addition, each of these problems can be classified as capacitated or uncapacitated, depending on various types of capacity restrictions. In particular, there can be limitations on the total flow routed on a hub-hub link (Labbé and Yaman 2004) or on the amount of flow into the hub nodes (Ernst and Krishnamoorthy 1999). Other constraints can be taken into account such as hub congestion (Elhedhli and Hu 2005), stochasticity (Contreras et al. 2011b, Rostami et al. 2018, 2020), and timing considerations (Kara and Tansel 2001, Yaman et al. 2012). For overviews of hub location problems, we refer the reader to Alumur and Kara (2008), Campbell and O'Kelly (2012), and Contreras (2015).

From a methodological perspective, different models and algorithms have been developed for the uncapacitated SAHLP (USAHLP) and the capacitated SAHLP (CSHALP). The first mathematical model for the uncapacitated USAHLP was proposed by O'Kelly (1987) as a quadratic integer program. The quadratic structure has led to the development of various linearization techniques (see for instance, Campbell 1994). Skorin-Kapov et al. (1996), Ernst and Krishnamoorthy (1996) proposed mixed integer linear programming (MILP) formulations for the USAHLP based on a path and on a flow representation, respectively. These models have been extensively used in the SAHLP literature. In the CSAHLP, a capacity on the flow transiting through a hub was considered by Ernst and Krishnamoorthy (1999). Contreras et al. (2011c) proposed a branch-and-price algorithm based on the Ernst and Krishnamoorthy (1999) formulation, in which lower bounds were computed through Lagrangian relaxation. Correia et al. (2010b) extended the CSAHLP by also considering the hub capacities as decision variables. In addition to these two basic MILP reformulations, one of the recent advances in the solution of the SAHLP is the study by Meier and Clausen (2018), in which Euclidean distances in the data sets are exploited to yield a new linearization of the quadratic formulation. Meier and Clausen (2018) applied this linearization to both the capacitated and uncapacitated versions of the SAHLP, and solved instances with up to 200 nodes taken from the well-known Australian Post (AP) data set (Ernst and Krishnamoorthy 1996).

Hub location problems can also be classified according to how economies of scale are modeled. In the first class, which we refer to as homogeneous economies of scale, the discount factor $\alpha$ between all hub-hub connections is fixed, i.e., it is flow-independent. This assumption builds on the fact that increased amounts of flow between hub-hub connections lead to cheaper unit transportation costs. This assumption was relaxed by O'Kelly and Bryan (1998) who considered a different discount factor depending on the amount of flow on a connection. We refer to this second class as heterogeneous hub location problems. Campbell and O'Kelly (2012) argue that even though a flow-dependent cost structure has been recognized in early works, it has not attracted much attention and the mainstream of research has employed computationally attractive homogeneous economies of scale ideas. O'Kelly and Bryan (1998) assumed the cost function to be a piecewise linear function of the flow, each flow segment having different fixed and variable costs. Furthermore, the discount factor $\alpha$ is parameterized by an arc index. The authors developed a model based on path-based variables for a multi-allocation hub location problem (MAHLP). To obtain a tighter formulation, de Camargo et al. (2009) built a model containing variables associated with each cost segment and path combination. Due to the large size of the model, the authors developed a Benders decomposition algorithm. Podnar et al. (2002), Cunha and Silva (2007) propose a different flow-dependent economies of scale in which the fixed cost discount factor $\alpha$ applies only when flow on the link exceeds a given threshold. This will allow either large or small vehicles to be used on any link,

based on the level of flow on that link. Vehicle-based and modular arc capacities are alternative approaches (see, for instance, Rostami et al. 2015, Tanash et al. 2017) in hub networks that model the economies of scale based on the number of used vehicles and installed links on all spoke-hub and hub-hub connections. For an overview on modeling economies of scale in hub location problem, we refer the reader to Alumur et al. (2020) and references therein.

### 1.1. Scientific Contributions

We introduce, model and solve the SAHLP with heterogeneous economies of scale (SAHLP-h), which is a generalization of the classical SAHLP. Heterogeneous economies of scales yield cost matrices that do not satisfy triangular inequality. Therefore the classical flow-based formulation of Ernst and Krishnamoorthy (1996), which is typically considered as the most effective formulation from a computational point of view (see, e.g., Correia et al. 2010a) cannot be applied for the SAHLP-h.

Our main contributions are summarized as follows:

- We present a non-linear model for the SAHLP-h that can handle heterogeneous economies of scale. We reformulate this non-linear model into a MILP with the same structure as the classical flow-based formulation of Ernst and Krishnamoorthy (1996). We exploit the special structure of the MILP to develop an integer L-shaped decomposition method (Laporte and Louveaux 1993) to solve the MILP reformulation, embedded within a branch-and-bound (B&B) framework.
- We reformulate the MILP into a mixed integer quadratically constrained program (MIQCP) model and implement a Benders type decomposition scheme. We dualize a set of complicating constraints in the resulting integer subproblem and generate a Lagrangian function, which yields a tight lower bound and an effective decomposition of the subproblem. We develop linear dual functions to underestimate the integer subproblem, which helps us obtain optimality cuts with a convergence guarantee by solving a linear program. We also develop specialized polynomial-time algorithms to generate enhanced cuts.

The remainder of the paper is organized as follows. Section 2 presents the problem statement and previous formulations. Section 3 outlines our new MILP formulation and describes the integer L-shaped algorithm. Section 4 details the MIQCP model and presents Lagrangian-based Benders decomposition. To evaluate the efficiency of our models and algorithms, we perform extensive computational experiments on both USAHLP and CSAHLP instances derived from the classical AP dataset. Section 5 presents the results of our computational experiments, and Section 6 concludes the paper.

## 2. Problem Statement

Consider a directed graph $G = (N, A)$, where $N = \{1, \ldots, n\}$ is the set of nodes representing the origins and destinations, which can also be possible hub locations, and $A$ is the set of arcs consisting of direct links between node pairs. From this point on, we use indices $i, j, k, \ell \in N$ for nodes and drop the set notation for convenience. Let $w_{ij}$ be the amount of flow to be transported from node $i$ to node $j$, and let $d_{ij}$ be the distance from node $i$ to node $j$. We define $O_i = \sum_j w_{ij}$ and $D_i = \sum_j w_{ji}$ as the total outgoing flow from node $i$ and the total incoming flow to node $i$, respectively. For each $k$, $f_k$ represents the fixed setup cost for locating a hub at node $k$.

We define a path $p = (i, k, \ell, j)$ from an origin node $i$ to a destination node $j$ passing through hubs $k$ and $\ell$ in that order. We assume that at most one hub arc will be used in each path (see Campbell et al. (2005)). In other words, for each path $p = (i, k, \ell, j)$ with $k \neq \ell$, the traffic $w_{ij}$ is sent on arc $(i, k)$, is then routed through on the inter-hub connection $(k, \ell)$, and is finally delivered on arc $(\ell, j)$. If $k = \ell$, the traffic $w_{ij}$ flows on arc $(i, k)$ and then on arc $(k, j)$. There are three types of cost associated with each path $p$: the collection cost $\chi_{ik}$, the transfer cost $\mathcal{F}_{k\ell}$, and the distribution cost $\delta_{\ell j}$. We assume that $\chi_{ik}$, $\mathcal{F}_{k\ell}$, and $\delta_{\ell j}$ are concave functions of the total flow on arcs $(i, k), (k, \ell)$, and $(\ell, j)$, respectively. Note that because of the single allocation assumption, the values of the collection and distribution functions $\chi_{ik}$, and $\delta_{\ell j}$ depend only on the flows $O_i$ and $D_j$, and hence can be determined a priori. For notational convenience, let $c_{ik} = \chi_{ik} + \delta_{ki}$ be the total collection and distribution cost between node $i$ and hub $k$. The amount of flow routed on the inter-hub connection $(k, \ell)$ is an output of the model. Let $v_{k\ell}$ be the amount of flow routed on the inter-hub connection $(k, \ell)$ with $k \neq \ell$ and $v_{kk} = 0$ for all $k$ since the cost associated with the flow between two spoke nodes that are assigned to the same hub node are captured by the $c_{ik}$ parameters. $\mathcal{F}_{k\ell}$ is then a concave function over the finite interval $[a_{k\ell}, b_{k\ell}]$, where $a_{k\ell}$ and $b_{k\ell}$ are the lower and upper bounds on the flow variable $v_{k\ell}$.

Following O'Kelly and Bryan (1998), Klincewicz (2002) and de Camargo et al. (2009), for each $k$ and $\ell$, we define the cost function $\mathcal{F}_{k\ell} : \bigcup_{s \in S} I_s \longrightarrow \mathbb{R}$ with $I_s = [L^s_{k\ell}, U^s_{k\ell}]$ as

$$\mathcal{F}_{k\ell}(v_{kl}) = \min_{s \in S} (\beta^s_{k\ell} + \alpha^s_{k\ell} v_{k\ell}), \tag{1}$$

where $\beta^s_{k\ell}$ and $\alpha^s_{k\ell}$ are the intercept and the slope of an affine function. In other words, $\beta^s_{k\ell}$ and $\alpha^s_{k\ell}$ are the fixed and variable costs of operating in segment $s$ on arc $(k, l)$, respectively. Note that the coefficients $\beta^s_{kl}$ are increasing with $s$ and the coefficient $\alpha^s_{kl}$ are decreasing with $s$. Figure 1 depicts the function on three segments of an arc $(k, l)$, showing the fixed and variable costs.

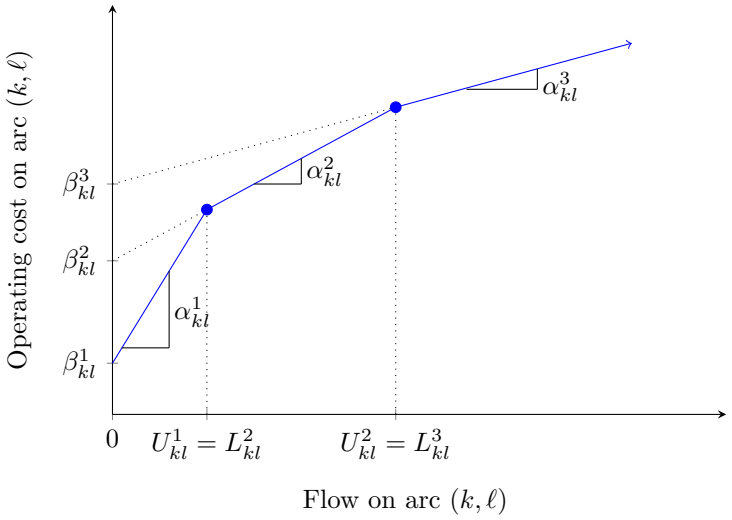


**Figure 1** Example of a cost function on arc $(k,\ell)$ with three segments, showing the fixed and variable costs.

The purpose of the SAHLP-h is to select a subset of nodes as hubs and to assign the remaining spoke nodes to these hubs so that each spoke node is assigned to exactly one hub while minimizing the sum of fixed setup, collection, distribution, and inter-hub connection costs. To formulate SAHLP-h, we introduce the variables

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

For every node $k$, $x_{kk}$ indicates whether $k$ is a hub ($x_{kk}=1$) or not ($x_{kk}=0$). We also define a binary variable $z^s_{k\ell}$ for each $k,\ell$ and $s \in S$ as follows:

$$z^s_{k\ell} = \begin{cases} 1 & \text{if the amount of flow between hubs } k \text{ and } \ell \text{ falls in the interval } I_s \\ 0 & \text{otherwise.} \end{cases}$$

For convenience, we refer to $x$ as hub location variables and to $z$ as flow segment variables. We use an $s$ index for the flow segment variables. Consequently, $v_{k\ell}$ and $\mathcal{F}_{k\ell}(v_{k\ell})$ can be expressed as

$$v_{k\ell} = \sum_i \sum_j w_{ij} x_{ik} x_{j\ell} \qquad \forall\ k,\ell, k \neq \ell, \qquad (2)$$

$$\mathcal{F}_{k\ell}(v_{k\ell}) = \sum_s z^s_{k\ell}(\beta^s_{k\ell} + \alpha^s_{k\ell}\, v_{k\ell}) = \sum_s z^s_{k\ell}(\beta^s_{k\ell} + \alpha^s_{k\ell} \sum_i \sum_j w_{ij} x_{ik} x_{j\ell}) \qquad \forall\ k,\ell, k \neq \ell, \qquad (3)$$

where (2) follows from the single allocation assumption, and (3) follows from the definition of the $z$ variable by which at most single flow segment variable can be non-zero on any hub-hub connection. The SAHLP-h can then be formulated as the following integer non-linear program:

SAHLP-h :

$$\text{minimize} \quad \sum_k f_k x_{kk} + \sum_i \sum_k c_{ik}\, x_{ik} + \sum_k \sum_\ell \sum_s d_{k\ell}\, \beta^s_{k\ell}\, z^s_{k\ell} + \sum_i \sum_j \sum_k \sum_\ell \sum_s d_{k\ell}\, \alpha^s_{k\ell} z^s_{k\ell} w_{ij} x_{ik} x_{j\ell} \tag{4}$$

subject to

$$\sum_k x_{ik} = 1 \qquad \forall\, i \tag{5}$$

$$x_{ik} \le x_{kk} \qquad \forall\, i,k \tag{6}$$

$$\sum_i O_i x_{ik} \le \Gamma_k\, x_{kk} \qquad \forall\, k \tag{7}$$

$$\sum_i \sum_j w_{ij} x_{ik} x_{j\ell} \ge \sum_s L^s_{k\ell}\, z^s_{k\ell} \qquad \forall\, k,\ell, k \ne \ell \tag{8}$$

$$\sum_i \sum_j w_{ij} x_{ik} x_{j\ell} \le \sum_s U^s_{k\ell}\, z^s_{k\ell} \qquad \forall\, k,\ell, k \ne \ell \tag{9}$$

$$\sum_s z^s_{k\ell} \ge x_{kk} + x_{\ell\ell} - 1 \qquad \forall\, k,\ell,\ k \ne \ell \tag{10}$$

$$x_{ik} \in \{0,1\} \qquad \forall\, i,k \tag{11}$$

$$z^s_{k\ell} \in \{0,1\} \qquad \forall\, k,\ell,s. \tag{12}$$

The objective is to minimize the total cost which includes three components: the cost of setting up the hubs, the cost of assigning the spoke nodes to the hub nodes, and the cost of transporting goods on hub-hub lines. Constraints (5) stipulate that each node $i$ is allocated to precisely one hub, while (6) enforce the condition that node $i$ is allocated to node $k$ only if $k$ is selected as a hub node. Constraints (7) restrict the total flow incoming to hubs. Constraints (8) and (9) ensure that the flow on each inter-hub arc $(k,\ell)$ lies within the interval $I_s$ if $z^s_{k\ell} = 1$. Constraints (10) force the activation of one segment $s$ for each arc $(k,\ell) \in A$ if both nodes $k$ and $\ell$ are selected as hub nodes. The domain restrictions are enforced by Constraints (11) and (12).

Such piecewise linear functions are represented using three main models, namely incremental, multiple choice and convex combination models (Croxton et al. 2003a). From a theoretical point of view, their LP relaxations are the same. However, the multiple choice model has been demonstrated to be computationally more efficient, particularly when the number of polytopes defining the piecewise linear function is small (Vielma et al. 2010). Therefore, the multiple choice model attracted greater attention in the literature (Croxton et al. 2003b, 2007, Frangioni and Gendron 2009). In Section EC.1 of the e-companion, we present the incremental and convex combination models and compare their computational performance with the multiple choice model, which is presented in Section 3.1. In the remainder, we use the multiple choice modeling framework.

For conciseness, we also define $\mathcal{X} = \{x \in \{0,1\}^{n^2} : (5), (6), (7)\}$ to be the set of solutions satisfying the location-allocation and the capacity constraints. If the problem is uncapacitated, we can simply relax (7).

## 3. A Linear Reformulation and an Integer L-shaped Method

The objective function of SAHLP-h is to minimize a cubic function over a non-convex set, which makes the problem intractable for any of the available solvers. In this section, we introduce a linearization of the SAHLP-h model and a Benders decomposition scheme as an exact solution technique.

The non-linearity of SAHLP-h stems from the quadratic terms $x_{ik}x_{j\ell}$, $i,j,k,\ell$, $k \neq \ell$ in (8) and (9), and from the cubic terms $z^s_{k\ell}x_{ik}x_{j\ell}$, $s,i,j,k,\ell$ in the third term in the objective function. In order to linearize the quadratic terms, one can apply a standard linearization technique by introducing new non-negative binary variables $r_{ikj\ell} = x_{ik}x_{j\ell}$ for $i,j,k,\ell$, and appending the following set of constraints to the model:

$$r_{ikj\ell} \leq x_{ik}, \quad r_{ikj\ell} \leq x_{j\ell}, \quad \text{and} \quad r_{ikj\ell} \geq x_{ik} + x_{j\ell} - 1 \qquad \forall\ i,j,k,\ell.$$

The same procedure can be repeated to linearize the new quadratic terms $r_{ikj\ell}z^s_{k\ell}$ in the objective function, which yields five-index variables. This standard linearization, however, does not take the problem structure into account and yields a weak reformulation. To overcome this difficulty, two main linearization techniques for the classical SAHLP were presented by Skorin-Kapov et al. (1996) and Ernst and Krishnamoorthy (1996, 1999). The resulting linearization models are known as the "path-based" and as the "flow-based" formulations, respectively. The path-based formulation uses $O(|V|^4)$ additional variables and $O(|V|^3)$ additional constraints to linearize the quadratic terms $x_{ik}x_{j\ell}$, while the flow-based model adds only $O(|V|^3)$ variables and $O(|V|^2)$ constraints. The flow-based formulation is widely considered as the most effective model for the classical SAHLP (see, for instance, Correia et al. 2010a). A crucial assumption for the validity of this linearization is that the triangle inequality for the transportation costs holds (Correia et al. 2010a). Considering the fact that the inter-hub connection costs are flow-dependent in our application, the triangular inequality for costs does not generally hold and therefore the classical flow-based technique cannot be applied to linearize the SAHLP-h. In what follows, we present a new flow-based model that always provides a valid linearization regardless of the underlying cost structure.

### 3.1. A new flow-based reformulation

To linearize the non-linearities in SAHLP-h, we define a new variable $y^s_{ik\ell}$ for each $i,k,\ell,s$ as the total amount of flow originating at node $i$ and routed via hubs located at nodes $k$ and $\ell$ using segment $s$. We refer to these variables as flow variables. Our linearization strategy relies on the following new equation

$$\sum_s y^s_{ik\ell} = x_{ik} \sum_j w_{ij}x_{j\ell} \qquad \forall\ i,k,\ell \tag{13}$$

that links the flow variables to the original $x$ variables. It is clear that $y^s_{ik\ell} = 0$ if node $i$ is not assigned to hub $k$ (i.e., $x_{ik} = 0$) and $\sum_s y^s_{ik\ell}$ is the total flow emanating from node $i$ and using hubs $k$ and $\ell$ (i.e., $\sum_j w_{ij}x_{j\ell}$) otherwise. We now present the necessary and sufficient conditions to linearize (13) in the following theorem.

THEOREM 1. *For $x \in \mathcal{X}$, the following two statements are equivalent:*

$(i)$ $$\sum_s y^s_{ik\ell} = x_{ik} \sum_j w_{ij}x_{j\ell} \quad \forall\ i,k,\ell$$

$(ii)$
$$\sum_s \sum_\ell y^s_{ik\ell} = O_i x_{ik}, \quad \forall\ i,k \tag{14}$$
$$\sum_s \sum_\ell y^s_{i\ell k} = \sum_j w_{ij}x_{jk}, \quad \forall\ i,k. \tag{15}$$

*Proof* For sufficiency, summing $\sum_s y^s_{ik\ell}$ over $\ell$ and $k$ and using $\sum_k x_{ik} = 1$, for all $i$, implies (14) and (15), respectively. For necessity, we assume *(ii)*. Using the fact that $\sum_k x_{ik} = 1$, for all $i$, let $a(i) = k$ if $x_{ik} = 1$. Now (14) implies

$$\sum_s \sum_\ell y^s_{ia(i)\ell} = O_i \qquad \forall\ i, \tag{16}$$
$$\sum_s y^s_{ik\ell} = 0 \qquad \forall\ k \neq a(i), i, \ell. \tag{17}$$

Then, (15) and (17) imply $\sum_s y^s_{ia(i)k} = \sum_j w_{ij}x_{jk}$. Summing over $k$ ensures that (16) also holds. Therefore,

$$\sum_s y^s_{ik\ell} = \begin{cases} \sum_j w_{ij}x_{j\ell} & \text{if } x_{ik} = 1 \\ 0 & \text{otherwise,} \end{cases}$$

which can also be expressed as $\sum_s y^s_{ik\ell} = x_{ik} \sum_j w_{ij}x_{j\ell}$, as desired. □

REMARK 1. With the introduction of the $y$ variables, constraints (8) and (9) can be disaggregated to yield the inequalities

$$\sum_i y^s_{ik\ell} \leq U^s_{k\ell} z^s_{k\ell} \quad \forall\ s,k,\ell,k \neq \ell \tag{18}$$
$$\sum_i y^s_{ik\ell} \geq L^s_{k\ell} z^s_{k\ell} \quad \forall\ s,k,\ell,k \neq \ell, \tag{19}$$

which ensure that the flow and segment variables are connected at the segment level.

Note that this disaggregation was not possible in (8) and (9). Introducing the $y^s_{ik\ell}$ variables enabled us to disaggregate these constraints. This disaggregation not only provides a strengthened formulation, but also leads to the following remark.

REMARK 2. In any feasible solution of the model obtained by appending (13) and (18) to SAHLP-h, we have $z^s_{k\ell}\, y^s_{ik\ell} = y^s_{ik\ell}$, for all $i,k,\ell,s$.

Furthermore, due to the concave cost structure, constraints (19) are not required. Then, SAHLP-h can be reformulated as the following flow-based MILP:

$$\begin{aligned}
&\text{MILP-flow:}\\
&\text{minimize} \quad \sum_k f_k x_{kk} + \sum_i \sum_k c_{ik}\, x_{ik} + \sum_s \sum_k \sum_\ell d_{k\ell}\, \beta^s_{k\ell}\, z^s_{k\ell} + \sum_s \sum_k \sum_\ell \sum_i d_{k\ell}\, \alpha^s_{k\ell}\, y^s_{ik\ell}\\
&\text{subject to} \quad x \in \mathcal{X}\\
&\qquad\qquad (10)-(12), (14), (15), (18)\\
&\qquad\qquad y^s_{ik\ell} \geq 0 \quad \forall\ i,k,\ell,s, \qquad (20)
\end{aligned}$$

where Constraints (14) and (15) are due to Theorem 1. Note that the MILP-flow model contains a polynomial number of variables and constraints and, hence, it can be solved directly through a MILP solver.

REMARK 3. Note that for $|S| = 1$, $\beta_{k\ell} = 0$, and $\alpha_{k\ell} = \alpha$, for all $k, \ell$, MILP-flow reduces to the classical SAHLP. Therefore, MILP-flow can also be considered as a new flow-based reformulation for the classical SAHLP regardless of the cost structure, and is valid for data that do not satisfy the triangular inequality.

### 3.2. Integer L-shaped Method

We now show how to exploit the special structure of MILP-flow which lends itself to decomposition techniques. The MILP-flow formulation naturally admits a Benders decomposition. In a classical application, we can project out the flow variables $y^s_{ikl}$ from the formulation and add Benders cuts to correctly estimate the distribution costs. However, one major drawback of such a procedure is that it requires feasibility as well as optimality cuts for convergence. Disconnecting the information exchange between the segment variables $z^s_{kl}$ and the flow variables $y^s_{ikl}$ by splitting them into a master problem and a subproblem results in a poor computational performance, as we observed in our preliminary implementations (see e-companion, Section EC.2). Keeping both the segment and flow variables in the subproblem, on the other hand, avoids feasibility cuts; however, the subproblem in that case is an integer linear programming model. To this end, we introduce an L-shaped method based on the MILP-flow.

We now partition the MILP-flow into an integer master problem and mixed-integer linear subproblems that are more manageable in size and computationally easier to solve with respect to the original model MILP-flow. The location-allocation variables $x$ are incorporated into the master problem, while the flow and segment variables $(y, z)$ are projected out and replaced by a single

variable $\xi$ representing the hub-hub transportation cost. The resulting master problem, which we refer to as MP1, is then given by

$$\begin{aligned} [\text{MP1}] \quad \text{minimize} \quad & \sum_k f_k x_{kk} + \sum_i \sum_k c_{ik}\, x_{ik} + \xi \\ \text{subject to} \quad & x \in \mathcal{X} \\ & \xi \geq \psi(x) \\ & x_{ik} \in \{0,1\} \quad \forall\ i,k, \end{aligned} \tag{21}$$

where $\psi(x)$ represents the hub-hub transportation cost for a given design $x \in \mathcal{X}$, provided by solving MILP slave problems. The decomposition idea is based on successively adding cuts in the $(x,\xi)$-space to approximate $\psi(x)$ until an optimal solution $(x^*, y^*, z^*)$ with $\xi = \psi(x^*)$ has been identified.

For a given $\hat{x} \in \mathcal{X}$, the slave problem, which we refer to as SP1($\hat{x}$), is obtained by fixing the $x$ variables to $\hat{x}$ in the MILP-flow formulation:

$$[\text{SP1}(\hat{x})] \quad \psi(\hat{x}) = \min \sum_s \sum_k \sum_\ell d_{k\ell}\, \beta^s_{k\ell}\, z^s_{k\ell} + \sum_s \sum_k \sum_\ell \sum_i d_{k\ell}\, \alpha^s_{k\ell}\, y^s_{ik\ell} \tag{22}$$

$$\text{subject to} \sum_s \sum_\ell y^s_{ik\ell} = O_i \hat{x}_{ik} \qquad \forall\ i,k \tag{23}$$

$$\sum_s \sum_\ell y^s_{i\ell k} = \sum_j w_{ij} \hat{x}_{jk} \qquad \forall\ i,k \tag{24}$$

$$\sum_i y^s_{ik\ell} \leq U^s_{k\ell}\, z^s_{k\ell} \qquad \forall\ k,\ell, k \neq \ell, s \tag{25}$$

$$\sum_s z^s_{k\ell} \geq \hat{x}_{kk} + \hat{x}_{\ell\ell} - 1 \qquad \forall\ k,\ell,\ k \neq \ell \tag{26}$$

$$z^s_{k\ell} \in \{0,1\} \qquad \forall\ k,\ell,s \tag{27}$$

$$y^s_{ik\ell} \geq 0 \qquad \forall\ i,k,\ell,s. \tag{28}$$

This subproblem is a MILP due to the integrality requirement of $z$ variables, which only admits integer L-shaped cuts (see Laporte and Louveaux 1993). Note that we do not explicitly need to add $\sum_s z^s_{kl} \leq 1$ constraint due to the sense of the objective function. Solving SP1 exactly provides us with a valid lower bound on $\xi$ that can be iteratively imposed by means of integer L-shaped cuts and the following integer optimality cut can be added to the MP1:

$$\xi \geq \psi(\hat{x}) \Big( 1 + \sum_{(i,k) \in \hat{A}} (x_{ik} - 1) - \sum_{(i,k) \in A \setminus \hat{A}} x_{ik} \Big), \tag{29}$$

where $\hat{A} = \{(i,k) \in A : \hat{x}_{ik} = 1\}$. Given that the cuts (29) are usually not strong, one can solve the linear programming (LP) relaxation of SP1($\hat{x}$) and add Benders optimality cuts to improve

the global lower bound on $\psi(x)$. As in Angulo et al. (2016), we implement a hybrid method, in which we first add Benders optimality cuts associated with the LP relaxation of SP1 to improve the bounds and resort to (29) to ensure optimality.

We have reformulated the SAHLP-h into a MILP model and we have developed an integer L-shaped method, which helped us transform a highly non-linear model into a more manageable one. These solution techniques constitute a basis for a Lagrangian-based Benders decomposition methodology, which we present in the following section. These techniques also serve as a basis for a computational comparison in our experiments in Section 5.

## 4. A quadratic reformulation and a Lagrangian-based Benders decomposition

We now develop a generalized Benders decomposition based on an extended reformulation and on a Lagrangian relaxation.

### 4.1. A New Reformulation and Decomposition

We append two sets of constraints to MILP-flow formulation to convert it into a mixed integer quadratically constrained programming model, which we refer to as MIQCP-flow:

MIQCP-flow:

$$\begin{aligned} \text{minimize} \quad & \sum_k f_k x_{kk} + \sum_i \sum_k c_{ik}\, x_{ik} + \sum_s \sum_k \sum_\ell d_{k\ell}\, \beta^s_{k\ell}\, z^s_{k\ell} + \sum_s \sum_k \sum_\ell \sum_i d_{k\ell}\, \alpha^s_{k\ell}\, y^s_{ik\ell} \\ \text{subject to} \quad & x \in \mathcal{X} \\ & (8)-(12), (14), (15), (18), (20). \end{aligned}$$

In this model, the quadratic constraints (8) and (9) are inherited from the SAHLP-h model for reasons that will become clear in Section 4.2. Our methodology relies on partitioning of MIQCP-flow, similar to what was done in Section 3.2, into an integer master problem and mixed integer subproblems. We introduce the variable $\eta$ to represent the hub-hub transportation cost. The resulting master problem, which we refer to as MP2, is then

$$\begin{aligned} \text{[MP2]} \quad \text{minimize} \quad & \sum_k f_k x_{kk} + \sum_i \sum_k c_{ik}\, x_{ik} + \eta \\ \text{subject to} \quad & x \in \mathcal{X} \\ & \eta \geq \phi(x) \qquad (30) \\ & x_{ik} \in \{0,1\} \quad \forall\ i,k, \end{aligned}$$

where $\phi(x)$ represents the hub-hub transportation cost for a given design $x \in \mathcal{X}$, provided by solving MILP slave problems. For a given $\hat{x} \in \mathcal{X}$, the slave problem, which we refer to as SP2($\hat{x}$), is obtained by fixing the $x$ variables to $\hat{x}$ in MIQCP-flow formulation:

$$[\text{SP2}(\hat{x})] \quad \phi(\hat{x}) = \min \sum_s \sum_k \sum_\ell d_{k\ell}\, \beta^s_{k\ell}\, z^s_{k\ell} + \sum_s \sum_k \sum_\ell \sum_i d_{k\ell}\, \alpha^s_{k\ell}\, y^s_{ik\ell} \tag{31}$$

$$\text{subject to} \quad (23)-(28)$$

$$\sum_i \sum_j w_{ij} \hat{x}_{ik} \hat{x}_{j\ell} \le \sum_s U^s_{k\ell}\, z^s_{k\ell} \quad \forall\ k, \ell, k \ne \ell \tag{32}$$

$$\sum_i \sum_j w_{ij} \hat{x}_{ik} \hat{x}_{j\ell} \ge \sum_s L^s_{k\ell}\, z^s_{k\ell} \quad \forall\ k, \ell, k \ne \ell. \tag{33}$$

This subproblem is a MILP due to the integrality requirement of the $z$ variables. In the following section, we show that dualizing a set of complicating constraints yields a tight Lagrangian relaxation and a generalized Benders decomposition.

### 4.2. A Tight Lagrangian Relaxation of the Subproblem

Here we provide an alternative method based on Lagrangian relaxation to obtain a tight approximation of $\phi(x)$. The idea of Lagrangian relaxation is to transfer one or more complicating constraints to the objective function as a penalty term, so that the resulting problem is an easier one or splits into a set of easier problems. We apply a Lagrangian relaxation of constraints (25) in SP2($\hat{x}$) using Lagrangian multipliers $\pi^s_{k\ell} \le 0,\ s, k, \ell, k \ne \ell$. The resulting Lagrangian function is:

$$\mathcal{L}(\hat{x}, \pi) = \min \sum_s \sum_k \sum_\ell (d_{k\ell}\, \beta^s_{k\ell} + U^s_{k\ell} \pi^s_{k\ell})\, z^s_{k\ell} + \sum_s \sum_k \sum_\ell \sum_i (d_{k\ell}\, \alpha^s_{k\ell} - \pi^s_{k\ell})\, y^s_{ik\ell}$$
$$\text{subject to} \quad (23), (24), (26)-(28), (32), (33).$$

Due to the choice of the relaxed constraints, the problem decomposes, so that the Lagrangian function can be written as $\mathcal{L}(\hat{x}, \pi) = \mathcal{L}^z(\hat{x}, \pi) + \mathcal{L}^y(\hat{x}, \pi)$ with

$$\mathcal{L}^z(\hat{x}, \pi) = \min_z \ \Big\{ \sum_s \sum_k \sum_\ell (d_{k\ell}\, \beta^s_{k\ell} + U^s_{k\ell} \pi^s_{k\ell})\, z^s_{k\ell} :\ (26), (27), (32), (33) \Big\}$$

and

$$\mathcal{L}^y(\hat{x}, \pi) = \min_y \ \Big\{ \sum_s \sum_k \sum_\ell \sum_i (d_{k\ell}\, \alpha^s_{k\ell} - \pi^s_{k\ell})\, y^s_{ik\ell} :\ (23), (24), (28) \Big\},$$

where $\mathcal{L}^z(\hat{x}, \pi)$ and $\mathcal{L}^y(\hat{x}, \pi)$ are independent subproblems in the $z$ and $y$ variables, respectively. Observe that $\mathcal{L}^z(\hat{x}, \pi)$ is an integer program while $\mathcal{L}^y(\hat{x}, \pi)$ is a linear program in the $y$ variables. It is known that $\mathcal{L}(\hat{x}, \pi)$ provides a lower bound on $\phi(\hat{x})$. The best lower bound on $\phi(\hat{x})$, which we refer to as $\phi_{LR}(x)$, is obtained by solving the following Lagrangian dual problem:

$$\phi_{LR}(\hat{x}) = \max_{\pi \le 0} \mathcal{L}(\hat{x}, \pi),$$

which ensures that $\phi_{LP}(\hat{x}) \leq \phi_{LR}(\hat{x}) \leq \phi(\hat{x})$. There exist several schemes for determining an optimal or a near-optimal solution $\pi$ (see Fisher 2004). However, since our solution method is an iterative procedure, solving the slave problem using such methods is computationally expensive. Instead, we take advantage of the problem structure to obtain an optimal $\pi$, for which $\phi_{LR}(\hat{x}) = \phi(\hat{x})$. Before we prove this strong result, we need a sequence of intermediate steps. Let us first consider the subproblem $L^z(\hat{x}, \pi)$.

LEMMA 1. *For $x \in \mathcal{X}$ and $\pi \leq 0$, the following conditions characterize the feasible solutions of $L^z(\hat{x}, \pi)$:*

$$
\begin{array}{lll}
(i) & \text{if } \hat{x}_{kk} = 0 \text{ or } \hat{x}_{\ell\ell} = 0, & \Rightarrow z^s_{k\ell} = 0 \qquad \forall\ k, \ell, s, \\
(ii) & \text{if } \hat{x}_{kk} = 1,\ \hat{x}_{\ell\ell} = 1 \text{ and } L^s_{k\ell} < \sum_{i,j \in N} w_{ij} \hat{x}_{ik} \hat{x}_{j\ell} < U^s_{k\ell}, & \Rightarrow z^s_{k\ell} = 1 \qquad \forall\ k, \ell, s, \\
(iii) & \text{if } \hat{x}_{kk} = 1,\ \hat{x}_{\ell\ell} = 1 \text{ and } L^s_{k\ell} = \sum_{i,j \in N} w_{ij} \hat{x}_{ik} \hat{x}_{j\ell}, & \Rightarrow z^s_{k\ell} = 1 \text{ or } z^{s-1}_{k\ell} = 1 \quad \forall\ k, \ell, s \geq 2, \\
(iv) & \text{if } \hat{x}_{kk} = 1,\ \hat{x}_{\ell\ell} = 1 \text{ and } \sum_{i,j \in N} w_{ij} \hat{x}_{ik} \hat{x}_{j\ell} = U^s_{k\ell}, & \Rightarrow z^s_{k\ell} = 1 \text{ or } z^{s+1}_{k\ell} = 1 \quad \forall\ k, \ell, s \leq |S| - 1.
\end{array}
$$

*Proof* Observe that the subproblem $\mathcal{L}^z(\hat{x}, \pi)$, can be decomposed into the following $n^2$ subproblems $\mathcal{L}^z_{k\ell}(\hat{x}, \pi)$, for each $k, \ell \in N$:

$$
\begin{aligned}
\mathcal{L}^z_{k\ell}(\hat{x}, \pi) = \min \quad & \sum_s (d_{k\ell}\, \beta^s_{k\ell} + U^s_{k\ell} \pi^s_{k\ell})\, z^s_{k\ell} && (34) \\
\text{subject to} \quad & \sum_i \sum_j w_{ij} \hat{x}_{ik} \hat{x}_{j\ell} \leq \sum_s U^s_{k\ell}\, z^s_{k\ell} && (35) \\
& \sum_i \sum_j w_{ij} \hat{x}_{ik} \hat{x}_{j\ell} \geq \sum_s L^s_{k\ell}\, z^s_{k\ell} && (36) \\
& \sum_s z^s_{k\ell} \geq \hat{x}_{kk} + \hat{x}_{\ell\ell} - 1 && (37) \\
& z^s_{k\ell} \in \{0, 1\} \quad \forall\ s. && (38)
\end{aligned}
$$

When $\hat{x}_{kk} = 0$ or $\hat{x}_{\ell\ell} = 0$, we have $\hat{z}^s_{kl} = 0$ for all $k, \ell \in N$ due to constraints (6), (35) and (36) as in part *(i)*). When $\hat{x}_{kk} = \hat{x}_{\ell\ell} = 1$, we have $\sum_{s \in S} z^s_{kl} \geq 1$ due to (37). Among $s \in S$, there can be only one satisfying strictly the bound constraints (35) and (36) as given in part *(ii)*. When we have $\sum_i \sum_j w_{ij} \hat{x}_{ik} \hat{x}_{j\ell} = L^s_{k\ell}$ for $s \geq 2$, then segments $s-1$ and $s$ can satisfy the bound constraints, which gives part *(iii)*. The same argument can be used to part *(iv)*, which completes the proof. □

REMARK 4. The Lagrangian relaxation decomposes the slave problem into two independent subproblems, which in turn disconnects the relationship between $y^s_{ikl}$ and $z^s_{kl}$ variables. However, the quadratic constraints that we have appended to the MIQCP-flow formulation ensure that the same implications between variables are in place. In other words, the optimal values of the $z$ variables in both SP2($\hat{x}$) and the decomposed $L^z(\hat{x}, \pi)$ problems are the same.

We now turn to our main result:

THEOREM 2. *The Lagrangian relaxation is tight (i.e., $\phi_{LR}(\hat{x}) = \phi(\hat{x})$) for the following setting of Lagrangian multipliers:*

$$\hat{\pi}^s_{k\ell} = \begin{cases} 0 & \text{if } \hat{z}^s_{k\ell} = 1 \\ -d_{k\ell} & \text{otherwise} \end{cases} \qquad \forall\ k, \ell, s. \tag{39}$$

*Proof* Consider $\mathcal{L}^z(\hat{x}, \hat{\pi})$. Because of the parameter $\pi$ as given in (39), of the fact that the coefficients $\beta^s_{kl}$ are increasing with $s$, and due to Remark 4, the optimal solutions characterized by Lemma 1 are also optimal for SP($\hat{x}$).

Now consider $\mathcal{L}^y(\hat{x}, \hat{\pi})$, which can be decomposed into $n$ subproblems $\mathcal{L}^y_i(\hat{x}, \hat{\pi})$ based on $i \in N$. According to constraints (23), (24) and Theorem 1, we have

$$\sum_s \hat{y}^s_{ik\ell} = \hat{x}_{ik} \sum_j w_{ij} \hat{x}_{j\ell} \qquad \forall\ k, \ell, \tag{40}$$

which holds for both SP?($\hat{x}$) and $\mathcal{L}^y_i(\hat{x}, \hat{\pi})$. Given the facts that $0 \le \alpha^s_{k\ell} \le 1$, the coefficients $\beta^s_{kl}$ are increasing with $s$ and the coefficient $\alpha^s_{kl}$ are decreasing with $s$, the parameter $\pi$ as given in (39) ensures that solving $\mathcal{L}^y_i(\hat{x}, \hat{\pi})$ yields

$$\hat{y}^s_{ik\ell} = \begin{cases} \hat{x}_{ik} \sum_j w_{ij} \hat{x}_{j\ell} & \text{if } \hat{z}^s_{k\ell} = 1 \\ 0 & \text{otherwise} \end{cases} \qquad \forall\ k, \ell, s, \tag{41}$$

which is also optimal for SP($\hat{x}$) according to the linking constraints (25). Hence, $\phi(\hat{x}) = \phi_{LR}(\hat{x}) = \sum_s \sum_k \sum_\ell d_{k\ell} (\beta^s_{k\ell} \hat{z}^s_{k\ell} + \alpha^s_{k\ell} \sum_i \hat{y}^s_{ik\ell})$. This completes the proof. □

COROLLARY 1. *The two functions $\phi : \mathcal{X} \to \mathbb{R}$ and $\phi_{LR} : \mathcal{X} \to \mathbb{R}$ are equal, i.e., $\phi(\hat{x}) = \phi_{LR}(\hat{x})$, for any $\hat{x} \in \mathcal{X}$.*

Now that we have constructed the optimal $\hat{\pi}$ values. From this point on, we can set $\phi_{LR}(x) = \phi^z_{LR}(x) + \phi^y_{LR}(x)$ where $\phi^z_{LR}(x) = \mathcal{L}^z(x, \hat{\pi})$ and $\phi^y_{LR}(x) = \mathcal{L}^y(x, \hat{\pi})$. In the following, we show how to underestimate function $\phi(x) = \phi_{LR}(x)$ to compute optimality cuts.

### 4.3. New Optimality Cuts Based on Lagrangian Relaxation

In order to generate optimality cuts, we now describe a procedure to construct a lower approximation of $\phi(x)$. Let $\underline{\phi}^z_{LR}$ and $\underline{\phi}^y_{LR}$ be the lower approximating functions of $\phi^z_{LR}$ and $\phi^y_{LR}$, respectively. Replacing $\phi(x)$ in (30) with the function $\underline{\phi}^z_{LR} + \underline{\phi}^y_{LR}$, the optimality cut in the MP then becomes

$$\eta \ge \underline{\phi}^z_{LR}(x) + \underline{\phi}^y_{LR}(x) \qquad x \in \mathcal{X}. \tag{42}$$

Once $\underline{\phi}^z_{LR}(x)$ and $\underline{\phi}^y_{LR}(x)$ are computed for a given $x \in \mathcal{X}$, one can generate the optimality cut on the fly using callbacks when the MP is solved as a MILP by a branch-and-cut algorithm using a state-of-the-art commercial MILP solver. In the following two subsections, we describe how to find such the lower approximating functions.

**4.3.1. Computing $\underline{\phi}^z_{LR}(x)$ Corresponding to the Integer Subproblem** Consider the integer subproblem in the $z$ variables. Due to its non-convexity, the subgradient information cannot be used to find a valid approximation of $\phi^z_{LR}(x)$. Instead, we use the concept of dual functions (see, for instance, Wolsey 1981, Guzelsoy and Ralphs 2007, Ralphs and Hassanzadeh 2014). A dual function $\underline{\phi}^z_{LR}(x)$ of $\phi^z_{LR}$ is a function that bounds $\phi^z_{LR}$ from below, that is $\underline{\phi}^z_{LR}(x) \leq \phi^z_{LR}(x)$, for each $x \in \mathcal{X}$.

For $\hat{x} \in \mathcal{X}$ and $\hat{\pi}$ as constructed in Theorem 2, consider solving the MILP $\mathcal{L}^z_{kl}(\hat{x})$, which is presented in the proof of Lemma 1, using a B&B algorithm. For each $k, \ell$, the B&B tree either has one or two feasible nodes due to Lemma 1. Therefore, one needs to solve the LP relaxation of the model corresponding to each leaf node of the B&B tree to compute the dual function. In what follows, we derive the dual function corresponding to the B&B tree with single feasible solution (when $L^s_{k\ell} < \sum_{i,j\in N} w_{ij}\hat{x}_{ik}\hat{x}_{j\ell} < U^s_{k\ell}$ in Lemma 1). We also discuss the case with two feasible solutions in Remark 5, which does not impact the general method we present now.

Given $k, \ell$, we consider segment $s$ for which $L^s_{k\ell} < \sum_{i,j\in N} w_{ij}\hat{x}_{ik}\hat{x}_{j\ell} < U^s_{k\ell}$. Due to part (ii) of Lemma 1, there exists a single node in the branch-and-bound tree, which provides an integer solution, and therefore the solution provided is optimal. We refer to this node as *the optimal node*, in which $z^s_{kl}$ variables are fixed to their values as given in Lemma 1. Here, the $\hat{z}^s_{kl}$ constants represent the branching rules in the B&B tree. The LP relaxation associated with the optimal node, which we refer to as $\hat{\mathcal{L}}^z_{kl}(\hat{x})_{LP}$, is the following LP:

$$
\begin{aligned}
[\hat{\mathcal{L}}^z_{kl}(\hat{x})_{LP}] \quad \text{minimize} \quad & \sum_s (d_{k\ell}\,\beta^s_{k\ell} + U^s_{k\ell}\hat{\pi}^s_{k\ell})\, z^s_{k\ell} \\
\text{subject to} \quad & (35)-(37) \\
& z^s_{k\ell} = \hat{z}^s_{k\ell} \quad \forall\ s. \qquad (43)
\end{aligned}
$$

Let $u_{k\ell}, \rho_{k\ell}, \psi_{k\ell}$ be the non-negative dual variables associated with $(35)-(37)$, respectively and let $\epsilon^s_{k\ell}$ be the dual variable, unrestricted in sign, associated with constraints (43). The dual of the LP at the optimal node, which we refer to as $\mathcal{D}^z_{k\ell}(\hat{x})_{LP}$, is then given by

$$
\begin{aligned}
[\mathcal{D}^z_{k\ell}(\hat{x})_{LP}] \quad \text{maximize} \quad & \sum_{i,j} w_{ij}\hat{x}_{ik}\hat{x}_{j\ell}(u_{k\ell} - \rho_{k\ell}) + \psi_{k\ell}(\hat{x}_{kk} + \hat{x}_{\ell\ell} - 1) + \sum_s \epsilon^s_{k\ell}\hat{z}^s_{k\ell} \\
\text{subject to} \quad & U^s_{k\ell}u_{k\ell} - L^s_{k\ell}\rho_{k\ell} + \psi_{k\ell} + \epsilon^s_{k\ell} \leq d_{k\ell}\beta^s_{k\ell} + U^s_{k\ell}\hat{\pi}^s_{k\ell} \qquad \forall\ s \\
& u_{k\ell}, \rho_{k\ell}, \psi_{k\ell} \geq 0.
\end{aligned}
$$

In the following theorem, we show that, using the optimal solution of $\mathcal{D}^z_{k\ell}(\hat{x})_{LP}$, we can compute a function $\underline{\phi}^z_{LR}(x)$, which is a lower approximating function of $\phi^z_{LR}(x)$.

THEOREM 3. *For a given $\hat{x} \in \mathcal{X}$, let $\hat{u}_{k\ell}, \hat{\rho}_{k\ell}, \hat{\psi}_{k\ell}$, and $\hat{\epsilon}^s_{k\ell}$ be the optimal solution of $\mathcal{D}^z_{k\ell}(\hat{x})_{LP}$. Then, there exists a function $\underline{\phi}^z_{LR}(x) = \sum_{k,\ell} \underline{\phi}^z_{k\ell}(x)$ such that $\underline{\phi}^z_{LR}(x) \le \phi^z_{LR}(x)$ for all $x \in \mathcal{X}$ and $\underline{\phi}^z_{LR}(\hat{x}) = \phi^z_{LR}(\hat{x})$, where*

$$\underline{\phi}^z_{k\ell}(x) = \sum_{i,j} w_{ij} x_{ik} x_{j\ell} (\hat{u}_{k\ell} - \hat{\rho}_{k\ell}) + \hat{\psi}_{k\ell}(x_{kk} + x_{\ell\ell} - 1) + \sum_s \hat{\epsilon}^s_{k\ell} \hat{z}^s_{k\ell}, \quad x \in \mathcal{X}. \tag{44}$$

*Proof* Theorem 3 directly follows from Theorem 2 of Ralphs and Hassanzadeh (2014), which is a reformulation of a result in Wolsey (1981). We omit the proof for the sake of conciseness. □

Note that the dual function $\underline{\phi}^z_{LR}(x)$ is a quadratic function of $x$. However, the following proposition shows how to reduce it to a linear function.

PROPOSITION 1. *Given $k$ and $\ell$, an optimal solution of $\mathcal{D}^z_{k\ell}(\hat{x})_{LP}$ can be found as follows:*

$$\begin{cases} \hat{u}_{k\ell} = \hat{\rho}_{k\ell} = \epsilon^{\hat{s}}_{k\ell} = 0 & \\ \hat{\psi}_{k\ell} = d_{k\ell} \beta^{\hat{s}}_{k\ell} & \text{if } \hat{x}_{kk} = \hat{x}_{\ell\ell} = 1 \\ \hat{\epsilon}^s_{k\ell} = d_{k\ell}(\beta^s_{k\ell} - U^s_{k\ell}) - \hat{\psi}_{k\ell} \quad \forall\ s \neq \hat{s} & \end{cases}$$

*and*

$$\begin{cases} \hat{u}_{k\ell} = \hat{\rho}_{k\ell} = \hat{\psi}_{k\ell} = 0 & \\ \hat{\epsilon}^s_{k\ell} = d_{k\ell}\, \beta^s_{k\ell} + U^s_{k\ell} \hat{\pi}^s_{k\ell} \quad \forall\ s & \text{if } \hat{x}_{kk} = 0 \text{ or } \hat{x}_{\ell\ell} = 0. \end{cases}$$

*Proof* The optimality derives from the feasibility of the primal and dual solutions for their problems and from the fact that the primal cost $\sum_s d_{k\ell} \beta^s_{k\ell}\, \hat{z}^s_{k\ell} = d_{k\ell} \beta^{\hat{s}}_{k\ell}$ is equal to the dual cost $\hat{\psi}_{k\ell}(\hat{x}_{kk} + \hat{x}_{\ell\ell} - 1) = d_{k\ell} \beta^{\hat{s}}_{k\ell}$ for $\hat{x}_{kk} = \hat{x}_{\ell\ell} = 1$. □

Using this optimal dual solution in (44), we obtain a linear function $\underline{\phi}^z_{LR}(x) = \sum_{k,\ell} \underline{\phi}^z_{k\ell}(x)$, which is a lower approximation of the integer subproblem.

REMARK 5. Note that, for each $k, \ell$, if the flow $\sum_{i,j \in N} w_{ij} \hat{x}_{ik} \hat{x}_{j\ell}$ is equal to either bounds $L^s_{k\ell}$ or $U^s_{k\ell}$ for some $s$, the MILP $\mathcal{L}^z_{kl}(\hat{x})$ would have two feasible solutions due to Lemma 1. Therefore, one needs to consider the LP relaxations in the nodes of the decision tree corresponding to both feasible solutions (which we refer to as optimal and non-optimal nodes, respectively) to derive the dual function (see Ralphs and Hassanzadeh 2014). However, the cut (42) we obtain from the optimal node is dominated by the cut obtained from the non-optimal node. This is because of Lemma 1, Theorem 2, and the fact that $\beta^{\hat{s}}_{k\ell} < \beta^{\hat{s}+1}_{k\ell}$. Therefore, considering (44) is satisfactory to guarantee convergence of the algorithm.

**4.3.2. Computing $\underline{\phi}^y_{LR}(x)$ Corresponding to the Continuous Subproblem** In order to generate the function $\underline{\phi}^y_{LR}(x)$ in (42), let us consider the dual of the subproblem on $y$ variables, which we refer to as $\mathcal{D}^y(\hat{x})$. The dual subproblem $\mathcal{D}^y(\hat{x})$ can be decomposed to $|N|$ subproblem $\mathcal{D}^y_i(\hat{x})$, for each $i \in N$ described as follows:

$$[\mathcal{D}^y_i(\hat{x})] \quad \text{maximize} \quad \sum_k O_i \hat{x}_{ik} \lambda_{ik} + \sum_k \sum_j w_{ij} \hat{x}_{jk} \mu_{ik} \tag{45}$$

$$\text{subject to} \quad \lambda_{ik} + \mu_{il} \leq d_{k\ell} \alpha^s_{k\ell} - \hat{\pi}^s_{kl} \qquad \forall\, k, \ell, s \tag{46}$$

$$\lambda, \mu \text{ unrestricted.} \tag{47}$$

Due to construction of the $\pi$ variables in Theorem 2, all constraints corresponding to segments for which $\hat{z}^s_{kl} = 0$ are trivially satisfied. For each $k, \ell$ with $\hat{x}_{kk} = 1$, $\hat{x}_{\ell\ell} = 1$, there exists a single constraint of type (46) associated with segment $\hat{s}$ such that $z^{\hat{s}}_{kl} = 1$ and the corresponding $\hat{\pi}^{\hat{s}}_{k\ell} = 0$. Let $\hat{\alpha}_{kl} = \alpha^{\hat{s}}_{kl}$. Then $\mathcal{D}^y_i(\hat{x})$ is given by

$$[\mathcal{D}^y_i(\hat{x})] \quad \text{maximize} \quad \sum_k O_i \hat{x}_{ik} \lambda_{ik} + \sum_k \sum_j w_{ij} \hat{x}_{jk} \mu_{ik} \tag{48}$$

$$\text{subject to} \quad \lambda_{ik} + \mu_{il} \leq \hat{\alpha}_{k\ell} d_{k\ell} \qquad \forall\, k, \ell \tag{49}$$

$$\lambda, \mu \text{ unrestricted.} \tag{50}$$

Let $\hat{\lambda}_{ik}, \hat{\mu}_{ik}$ be the optimal solution of $\mathcal{D}^y_i(\hat{x})$. Then, by LP duality, we can set

$$\underline{\phi}^y_{LR}(x) = \sum_k O_i x_{ik} \hat{\lambda}_{ik} + \sum_k \sum_j w_{ij} x_{jk} \hat{\mu}_{ik}. \tag{51}$$

### 4.4. Cut enhancement via a two-phase algorithm

We now present an enhancement of the generated cut. For $\hat{x} \in \mathcal{X}$, the optimal solution of $\mathcal{L}^y_i(\hat{x})$ is unique due to (41). However the dual problem can possibly have multiple optimal solutions due to degeneracy. To this end, we further investigate the properties of the optimal solutions of the dual subproblem. The following lemma and proposition formally presents the conditions, under which a solution of $\mathcal{L}^y_i(\hat{x})$ is optimal.

LEMMA 2. *For a given $i, k, \ell \in N$ and $\hat{x} \in \mathcal{X}$, the following three statements are equivalent:*

*(i) $\hat{x}_{ik} > 0$ and $\sum_j w_{ij} \hat{x}_{j\ell} > 0$,*

*(ii) $y^{\hat{s}_{k\ell}}_{k\ell} > 0$ in an optimal solution of $\mathcal{L}^y_i(\hat{x})$,*

*(iii) $\lambda_{ik} + \mu_{il} = \hat{\alpha}_{k\ell} d_{k\ell}$ in an optimal solution of the $\mathcal{D}^y_i(\hat{x})$.*

*Proof* (i) $\Leftrightarrow$ (ii) due to Equation (13). Due to complementary slackness conditions between $\mathcal{L}^y_i(\hat{x})$ and $\mathcal{D}^y_i(\hat{x})$, we have (ii) $\Leftrightarrow$ (iii). Hence the result follows. $\square$

PROPOSITION 2. *For a given $i \in N$ and $\hat{x} \in \mathcal{X}$, $(\lambda^*, \mu^*) \in \Omega$ is an optimal solution of $\mathcal{D}_i^y(\hat{x})$ if and only if the following conditions are satisfied:*

*(i)* $\lambda_{ik}^* + \mu_{i\ell}^* = \hat{\alpha}_{k\ell} d_{k\ell}$ *if* $\hat{x}_{ik} = 1$ *and* $\hat{x}_{\ell\ell} = 1$,

*(ii)* $\lambda_{ik}^* + \mu_{i\ell}^* \leq \hat{\alpha}_{k\ell} d_{k\ell}$ *otherwise.*

*Proof* (Sufficiency) Assume that the solution $(\lambda^*, \mu^*)$ is optimal for $\mathcal{D}_i^y(\hat{x})$. Part *(ii)* follows from the feasibility of the solution. Due to Lemma 2, we have $\lambda_{ik}^* + \mu_{i\ell}^* = \hat{\alpha}_{k\ell} d_{k\ell}$ for all $k, \ell \in N$, $\hat{x}_{ik} > 0$ and $\sum_j w_{ij} \hat{x}_{j\ell} > 0$. Because $\hat{x} \in \mathcal{X}$, we have $\hat{x} \in \{0,1\}^{|N|^2}$ and therefore $\hat{x}_{ik} > 0$ implies $\hat{x}_{ik} = 1$. Having $\sum_j w_{ij} \hat{x}_{j\ell} > 0$ also implies that $\ell$ is a hub node and hence $\hat{x}_{\ell\ell} = 1$. In other words, the condition essentially translates into $\hat{x}_{ik} = 1$ and $\hat{x}_{\ell\ell} = 1$, as given in part *(i)*.

(Necessity) Assume that conditions *(i)* and *(ii)* are satisfied in a solution. Such a solution is trivially feasible for $\mathcal{D}_i^y(\hat{x})$. Next, we show that all solutions satisfying these two conditions assume the same objective function value, which is therefore the optimal value. First, we note that having $\hat{x} \in \mathcal{X}$ requires $\sum_{k \in N} \hat{x}_{ik} = 1$, which implies that there exists an index $a(i)$ with $\hat{x}_{ia(i)} = 1$. Put differently, $a(i)$ is the hub that node $i$ is assigned to or $i$ is a hub node (i.e., $i = a(i)$). Using this notation, the terms in the objective function of $\mathcal{D}_i^y(\hat{x})$ can be reorganized as follows:

$$\sum_k O_i \hat{x}_{ik} \lambda_{ik} + \sum_\ell \sum_j w_{ij} \hat{x}_{j\ell} \mu_{i\ell} = \sum_j w_{ij} \sum_k \hat{x}_{ik} \lambda_{ik} + \sum_j w_{ij} \sum_\ell \hat{x}_{j\ell} \mu_{i\ell} = \sum_j w_{ij} (\lambda_{ia(i)} + \mu_{ia(j)})$$

Since $x_{ia(i)} = 1$ and having $x_{ia(j)} = 1$ requires $x_{a(j)a(j)} = 1$ for all $j \in N$, we have

$$\lambda_{ia(i)} + \mu_{ia(j)} = \hat{\alpha}_{a(i)a(j)} d_{a(i)a(j)}$$

due to *(i)*. Hence, for any feasible solution of the $i^{th}$ subproblem $\mathcal{D}_i^y(\hat{x})$, the objective function value equals $\sum_j w_{ij} \hat{\alpha}_{a(i)a(j)} d_{a(i)a(j)}$, and therefore, any feasible solution for $\mathcal{D}_i^y(\hat{x})$ is also optimal. □

Proposition 2 characterizes the optimal solutions of the dual subproblem $\mathcal{D}_i^y(\hat{x})$. It is therefore crucial to select those that help generate 'enhanced' cuts to reduce the number of total iterations. To this end, in the following we will describe an enhancement strategy inspired by Pareto-optimal cut generation scheme of Magnanti and Wong (1981). The algorithm uses an interior point of the feasible region of the master problem in contrast to Kelley's cutting plane method, where the solution provided by the master problem is an extreme point (see Kelley 1960). Moreover, it ensures the optimality of the solution for the dual subproblem $\mathcal{D}_i^y(\hat{x})$. Let us consider the subproblem $\mathcal{D}_i^y(\hat{x})$. Given $\bar{x} \in ri(conv(\mathcal{X}))$, where $ri(conv(\mathcal{X}))$ is the relative interior of the convex hull of the feasible region of the master problem, we can solve the linear program below to generate alternative optimal solution of $\mathcal{D}_i^y(\hat{x})$:

$$[\overline{\mathcal{D}}_y^i(\hat{x}, \bar{x})] \quad \text{maximize} \quad \sum_k O_i \bar{x}_{ik} \lambda_{ik} + \sum_k \sum_j w_{ij} \bar{x}_{jk} \mu_{ik} \tag{52}$$

$$\text{subject to} \quad \lambda_{ik} + \mu_{i\ell} = \hat{\alpha}_{k\ell} d_{k\ell} \qquad \forall\ k, \ell : \hat{x}_{ik} = 1 \text{ and } \hat{x}_{\ell\ell} = 1 \qquad (53)$$

$$\lambda_{ik} + \mu_{i\ell} \leq \hat{\alpha}_{k\ell} d_{k\ell} \qquad \forall\ k, \ell : \hat{x}_{ik} \neq 1 \text{ or } \hat{x}_{\ell\ell} \neq 1, \qquad (54)$$

where according to Proposition 2, constraints (53) and (54) ensure the feasibility and optimality of $(\lambda, \mu)$. The $\overline{\mathcal{D}}_y^i(\hat{x}, \bar{x})$ model contains $|N|^2$ constraints. Note that, differently from $\mathcal{D}_y^i(\hat{x})$, this problem does not have the structure of the dual of the transportation problem. Explicitly building and solving it for each $i \in N$ is costly in terms of solution times, especially for instances with a large number of nodes. To avoid this, we devise a simple yet efficient algorithm with an $\mathcal{O}(n \log n)$ run time complexity that generates a feasible solution for $\overline{\mathcal{D}}_y^i(\hat{x}, \bar{x})$ without building and solving an LP.

**Algorithm 1:** Two-phase Algorithm (2PA)

**Input**: $w, d, \hat{x}, \bar{x}, i$.

**Output**: A feasible solution for $\overline{\mathcal{D}}_y^i(\hat{x}, \bar{x})$

**Define** $variable(type, row\_idx, col\_idx, corePt, value)$; // Initialization

**Init** $L \leftarrow$ new list();

**for each** $k \in N$ **do**

  $\lambda_{ik} \leftarrow$ new $variable(\lambda, i, k, O_i \bar{x}_{ik}, null)$;

  $\mu_{ik} \leftarrow$ new $variable(\mu, i, k, \sum_{j \in N} w_{ij} \bar{x}_{jk}, null)$;

  $L \leftarrow L \cup \{\lambda_{ik}, \mu_{ik}\}$;

$a(i) \leftarrow$ **Select** $k \in N : \hat{x}_{ik} = 1$; // Phase 1

$\lambda_{ia(i)}.value \leftarrow 0$;

**for each** $\ell \in N$ **do**

  **if** $\hat{x}_{\ell\ell} = 1$ **then** $\mu_{i\ell}.value \leftarrow \hat{\alpha}_{k\ell} d_{a(i)\ell}$;

$L$.sort(key = corePt); // Phase 2

**while** $L \neq \emptyset$ **do**

  **Set** $var = L.pop()$;

  **if** $var.value = null$ **then**

    **if** $var.type = \mu$ **then**

      $i \leftarrow var.row\_idx$; $\ell \leftarrow var.col\_idx$;

      $\mu_{il}.value = \min_{k \in N : \lambda_{ik}.value \neq null} \{\hat{\alpha}_{k\ell} d_{k\ell} - \lambda_{ik}\}$;

    **else**

      $i \leftarrow var.row\_idx$; $k \leftarrow var.col\_idx$;

      $\lambda_{ik}.value = \min_{\ell \in N : \mu_{i\ell}.value \neq null} \{\hat{\alpha}_{k\ell} d_{k\ell} - \mu_{i\ell}\}$;

**Return** $(\lambda, \mu)$

The two-phase algorithm (2PA) is presented in Algorithm 1. The inputs of the algorithm are a flow matrix $w$, a unit flow cost matrix $d$, a fixed solution $\hat{x}$, core point $\bar{x}$ and $i \in N$, and the output is

a feasible solution for $\overline{\mathcal{D}}_y^i(\hat{x},\bar{x})$. The algorithm contains two phases and starts with an initialization step which creates objects required for the two phase algorithm to run. First, we introduce a new object called *variable*. It has the following properties: the variable type ($type \in \{\lambda,\mu\}$), the row and column indices (*row_idx* and *col_idx*), the objective function coefficient of the variable in $\overline{\mathcal{D}}_y^i(\hat{x},\bar{x})$ model (*corePt*) and the value of the variable when the algorithm terminates (*value*). This *value* is initially set as *null*, implying that no value has yet been assigned. Lines 4 and 5 create the variables which are added to a list called $L$. Phase 1 of the algorithm in lines 7−10 coincides with part *(i)* of Proposition 2 and assigns dual variable values when $\hat{x}_{ik}=1$ and $\hat{x}_{kk}=1$. This assignment ensures that $\lambda^*_{ik}+\mu^*_{il}=\hat{\alpha}_{k\ell}d_{k\ell}$. Phase 2 in lines 11−20 coincides with part *(ii)* of Proposition 2. The while loop assigns values in a greedy fashion to those variables that have not been considered in the first phase by ensuring that $\lambda^*_{ik}+\mu^*_{il}\leq\hat{\alpha}_{k\ell}d_{k\ell}$, as required in part *(ii)* of Proposition 2. The following proposition is an immediate result of this procedure:

PROPOSITION 3. *The solution returned by the two-phase algorithm is optimal for* $\mathcal{D}_i^y(\hat{x})$.

*Proof* Since the solution generated by the 2PA satisfies Proposition 2, it is optimal for $\mathcal{D}_i^y(\hat{x})$. □

The algorithm assigns values to the variables in a greedy fashion. Starting with the variable having the largest objective function coefficient in $\overline{\mathcal{D}}_y^i(\hat{x},\bar{x})$, the algorithm in Phase 2 always assigns the maximum value that a variable can take. In this regard, even though the solution generated by the algorithm is not guaranteed to be optimal for the $\overline{\mathcal{D}}_y^i(\hat{x},\bar{x})$, our experimental results show that the algorithm is highly effective in finding the optimal objective function value.

## 5. Computational experiments

In this section, we present the computational experiments we have carried out to evaluate the empirical performance of the proposed methodology. The MILP-flow model developed in Section 3 is solved using Gurobi 8.0.1. For the Benders decompositions, we solve the MP-1 and MP-2 models as MILPs using a branch-and-cut algorithm in which the optimality cuts are generated on the fly using callbacks. We refer to the branch-and-cut algorithm based on hybrid method in Section 3.2 as *BC-LS*. We also denote the branch-and-cut algorithm based on Lagrangian relaxation based cut (42) as *BC-LR*. When we use the cut enhancement using 2PA algorithm, we then refer to the algorithm as *BC-LR+*.

We performed the experiments on the Calcul Québec computing infrastructure with Intel Xeon X5650 @ 2.67 GHz processors and a memory limit of 90 GB. All experiments were executed in sequential form using one thread with a time limit of 2 hours to solve each instance. All algorithms were implemented in C++ using the Gurobi 8.0.1 callback library.

**Table 1** Characteristics of the hub-hub concave cost function as proposed in Klincewicz (2002)

| Interval | Flow on | Modest scenario | | Median scenario | | Aggressive scenario | |
|---|---|---|---|---|---|---|---|
| $s$ | link $(k,\ell)$ | $\alpha^s_{k\ell}$ | $\beta^s_{k\ell}$ | $\alpha^s_{k\ell}$ | $\beta^s_{k\ell}$ | $\alpha^s_{k\ell}$ | $\beta^s_{k\ell}$ |
| 1 | $[0,50)$ | 1.0 | 0.0 | 1.0 | 0.0 | 0.8 | 0.0 |
| 2 | $[50,100)$ | 0.9 | 5.0 | 0.8 | 10.0 | 0.6 | 10.0 |
| 3 | $[100,200)$ | 0.8 | 15.0 | 0.6 | 30.0 | 0.4 | 30.0 |
| 4 | $200\leq$ | 0.7 | 35.0 | 0.4 | 70.0 | 0.2 | 70.0 |

In what follows, we first present test instances in Section 5.1. We provide the results of evaluating the MILP-flow model in Section 5.2, the performance of the proposed algorithms in terms of computational efforts in Section 5.3, and the detailed results of the BC-LR+ in Section 5.4.

### 5.1. Test instances

We tested the algorithms on a modified AP data set (Ernst and Krishnamoorthy 1996, 1999) which consists of instances having between 10 and 200 nodes. Two types of instances denoted by L and T were tested for the uncapacitated case with type T having a higher fixed costs for the nodes with large flows, while type L do not exhibit this feature. For the capacitated case, four types of instances denoted by LL, LT, TL, and TT were tested, where the first letter denotes the fixed cost type similar to the uncapacitated case, while the second letter indicates tight (T) and loose (L) node capacities. We used three piecewise linear concave cost functions as in Klincewicz (2002) and de Camargo et al. (2009) each consisting of the lower envelope of four affine functions with different intercepts and slopes. These three functions correspond to different levels of benefits from the economies of scale: modest, median, and aggressive as shown in Table 1. In the following subsections we denote by AP I, AP II, and AP III, the AP dataset under modest, median, and aggressive scenarios, respectively.

### 5.2. Performance of MILP-flow formulation

This section provides detailed computational results of the MILP-flow model. Tables 2 and 3 report the results for the uncapacitated SAHLP-h (USAHLP-h) and capacitated SAHLP-h (CSAHLP-h), respectively. In Table 2, the first column provides the number of nodes (nodes). The second column shows the cost structure (L/T). For each dataset AP I, AP II, and AP III, we report the best upper bound (UB) found within the time limit, the solution time in seconds of the LP relaxation ($T_{LP}$), as well as the total solution time of the model ($T_{IP}$). If the solver was not able to solve the LP or IP models for a given instance within the time limit, the time limit in seconds is reported as the solution time. Moreover, for each dataset, we also report the LP gap ($G_{LP}$) and the optimality gap ($G_{IP}$) in percentage. The LP gap is calculated as $G_{LP} = (UB - LB)/UB \times 100$, where LB is

**Table 2** Performance of MILP-flow formulation on USAHLP-h instances

| | | AP I | | | | | AP II | | | | | AP III | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Time (s) | | Gap (%) | | | Time (s) | | Gap (%) | | | Time (s) | | Gap (%) | |
| Nodes | Cost | UB | $T_{LP}$ | $T_{IP}$ | $G_{LP}$ | $G_{IP}$ | UB | $T_{LP}$ | $T_{IP}$ | $G_{LP}$ | $G_{IP}$ | UB | $T_{LP}$ | $T_{IP}$ | $G_{LP}$ | $G_{IP}$ |
| 10 | L | 225928 | 0.1 | 0.4 | 1.94 | 0.00 | 220117 | 0.1 | 0.8 | 3.48 | 0.00 | 213652 | 0.1 | 0.7 | 3.67 | 0.00 |
| 10 | T | 264452 | 0.1 | 0.6 | 1.68 | 0.00 | 257614 | 0.1 | 0.4 | 2.43 | 0.00 | 251412 | 0.1 | 0.3 | 2.87 | 0.00 |
| 20 | L | 234585 | 1.1 | 3.3 | 0.55 | 0.00 | 228164 | 0.7 | 3.0 | 1.02 | 0.00 | 223111 | 1.0 | 3.2 | 1.04 | 0.00 |
| 20 | T | 271010 | 1.0 | 3.4 | 0.63 | 0.00 | 263845 | 1.0 | 3.5 | 1.08 | 0.00 | 258207 | 0.7 | 5.9 | 1.07 | 0.00 |
| 25 | L | 236650 | 1.4 | 8.4 | 0.82 | 0.00 | 230917 | 2.1 | 8.6 | 1.14 | 0.00 | 226155 | 2.1 | 15.5 | 1.05 | 0.00 |
| 25 | T | 295668 | 1.2 | 8.7 | 0.41 | 0.00 | 289172 | 1.1 | 14.7 | 1.09 | 0.00 | 283384 | 1.3 | 7.5 | 1.06 | 0.00 |
| 40 | L | 240934 | 6.9 | 96.6 | 1.21 | 0.00 | 234628 | 7.0 | 81.9 | 1.60 | 0.00 | 229500 | 5.6 | 62.9 | 1.67 | 0.00 |
| 40 | T | 293165 | 5.9 | 59.2 | 0.00 | 0.00 | 293165 | 7.0 | 106.1 | 0.12 | 0.00 | 290704 | 6.7 | 56.6 | 1.02 | 0.00 |
| 50 | L | 237478 | 28.6 | 348.9 | 0.73 | 0.00 | 231825 | 26.9 | 242.2 | 1.23 | 0.00 | 227204 | 24.2 | 241.9 | 1.72 | 0.00 |
| 50 | T | 300421 | 26.2 | 299.2 | 0.49 | 0.00 | 294551 | 14.3 | 216.7 | 1.29 | 0.00 | 288141 | 24.0 | 323.1 | 1.39 | 0.00 |
| 60 | L | 228885 | 31.4 | 521.2 | 0.88 | 0.00 | 222745 | 27.3 | 483.7 | 1.25 | 0.00 | 217723 | 44.1 | 753.0 | 1.21 | 0.00 |
| 60 | T | 264628 | 24.1 | 465.8 | 0.61 | 0.00 | 260273 | 27.2 | 935.7 | 1.06 | 0.00 | 255780 | 43.1 | 386.5 | 1.40 | 0.00 |
| 70 | L | 226181 | 56.1 | 2229.6 | 0.96 | 0.00 | 220060 | 45.8 | 1691.3 | 1.66 | 0.00 | 215019 | 46.6 | 1487.0 | 2.48 | 0.00 |
| 70 | T | 261295 | 42.8 | 1963.8 | 0.00 | 0.00 | 261295 | 38.8 | 2045.6 | 0.00 | 0.00 | 261295 | 41.7 | 1055.2 | 0.00 | 0.00 |
| 75 | L | 235756 | 87.4 | 3359.7 | 0.96 | 0.00 | 229630 | 118.4 | 2296.2 | 1.23 | 0.00 | 224719 | 67.8 | 2234.2 | 1.29 | 0.00 |
| 75 | T | 288778 | 107.1 | 1638.1 | 0.00 | 0.00 | 287687 | 60.8 | 2073.7 | 0.53 | 0.00 | 284805 | 59.7 | 3151.1 | 0.48 | 0.00 |
| 90 | L | 225512 | 178.3 | 5831.6 | 0.92 | 0.00 | 219494 | 143.8 | 7028.1 | 1.30 | 0.00 | 214440 | 223.6 | 6350.0 | 1.59 | 0.00 |
| 90 | T | 257477 | 117.0 | 5362.3 | 0.55 | 0.00 | 252975 | 113.8 | 4633.0 | 0.85 | 0.00 | 249242 | 196.2 | 5088.4 | 0.80 | 0.00 |
| 100 | L | N/A | 427.9 | 7200.0 | N/A | N/A | N/A | 221.6 | 7200.0 | N/A | N/A | N/A | 186.6 | 7200.0 | N/A | N/A |
| 100 | T | N/A | 291.6 | 7200.0 | N/A | N/A | N/A | 167.2 | 7200.0 | N/A | N/A | N/A | 155.8 | 7200.0 | N/A | N/A |
| 125 | L | N/A | 1491.8 | 7200.0 | N/A | N/A | N/A | 657.1 | 7200.0 | N/A | N/A | N/A | 602.5 | 7200.0 | N/A | N/A |
| 125 | T | N/A | 441.4 | 7200.0 | N/A | N/A | N/A | 430.9 | 7200.0 | N/A | N/A | N/A | 416.5 | 7200.0 | N/A | N/A |
| 150 | L | N/A | 1585.2 | 7200.0 | N/A | N/A | N/A | 1271.8 | 7200.0 | N/A | N/A | N/A | 1058.3 | 7200.0 | N/A | N/A |
| 150 | T | N/A | 1005.8 | 7200.0 | N/A | N/A | N/A | 789.8 | 7200.0 | N/A | N/A | N/A | 838.2 | 7200.0 | N/A | N/A |
| 175 | L | N/A | 3786.1 | 7200.0 | N/A | N/A | N/A | 4540.9 | 7200.0 | N/A | N/A | N/A | 1790.3 | 7200.0 | N/A | N/A |
| 175 | T | N/A | 3386.1 | 7200.0 | N/A | N/A | N/A | 1635.1 | 7200.0 | N/A | N/A | N/A | 2931.2 | 7200.0 | N/A | N/A |
| 200 | L | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 5082.0 | 7200.0 | N/A | N/A | N/A | 2560.8 | 7200.0 | N/A | N/A |
| 200 | T | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 6637.5 | 7200.0 | N/A | N/A |
| | | Average | 983.3 | 3364.3 | 0.74 | 0.00 | | 808.3 | 3352.3 | 1.24 | 0.00 | | 641.7 | 3329.4 | 1.43 | 0.00 |
| | | Geometric mean | 57.4 | 513.9 | | | | 48.2 | 528.5 | | | | 48.5 | 505.2 | | |

N/A: Not available due to time limit

the value of the LP relaxation. If a gap value is not obtained within the time limit, it is reported as 'N/A'. The average solution time and gaps are reported in the last two rows. The descriptions are the same for Table 3.

As shown in Table 2, the MILP model can be solved for L and T instances having up to 90 nodes. The integrality gaps for AP I, AP II, and AP III datasets are on average 0.74%, 1.24%, 1.43%, respectively. The average solution times of the instances solved to optimality are 3364.3 seconds, 3352.3 seconds, and 3329.4 seconds, respectively. The solution times increase rapidly with the size of the instances, and the instances with 100 or more nodes cannot be solved within the time limit. In any of these unsolved instances, Gurobi could not identify an integer feasible solution. Furthermore, for 200 nodes in AP I datase, the LP relaxation cannot even be solved. For the capacitated instances, in Table 3, the general behavior of the model is similar to that of the

**Table 3** Performance of MILP-flow formulation on CSAHLP-h instances

| | | AP I | | | | | AP II | | | | | AP III | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Cost/ | | Time (s) | | Gap (%) | | | Time (s) | | Gap (%) | | | Time (s) | | Gap (%) | |
| Nodes | Capacity | UB | LP | IP | LP | IP | UB | LP | IP | LP | IP | UB | LP | IP | LP | IP |
| 10 | LL | 225928 | 0.1 | 0.4 | 1.94 | 0.00 | 220117 | 0.1 | 0.8 | 3.48 | 0.00 | 213652 | 0.1 | 0.7 | 3.67 | 0.00 |
| 10 | LT | 255755 | 0.1 | 1.0 | 3.67 | 0.00 | 248992 | 0.1 | 0.8 | 5.16 | 0.00 | 241356 | 0.1 | 0.6 | 5.60 | 0.00 |
| 10 | TL | 264452 | 0.1 | 0.5 | 1.68 | 0.00 | 257614 | 0.1 | 0.8 | 2.43 | 0.00 | 251412 | 0.1 | 0.6 | 2.87 | 0.00 |
| 10 | TT | 264452 | 0.1 | 0.4 | 1.68 | 0.00 | 257614 | 0.1 | 0.3 | 2.17 | 0.00 | 252218 | 0.1 | 0.5 | 2.53 | 0.00 |
| 20 | LL | 234585 | 1.1 | 2.7 | 0.55 | 0.00 | 228164 | 0.7 | 2.9 | 1.02 | 0.00 | 223111 | 0.7 | 5.3 | 1.04 | 0.00 |
| 20 | LT | 253352 | 0.5 | 24.1 | 2.96 | 0.00 | 249067 | 1.0 | 21.3 | 5.08 | 0.00 | 245480 | 0.6 | 27.8 | 6.35 | 0.00 |
| 20 | TL | 271010 | 0.6 | 3.4 | 0.63 | 0.00 | 263845 | 0.7 | 3.6 | 1.08 | 0.00 | 258207 | 1.0 | 5.9 | 1.07 | 0.00 |
| 20 | TT | 298498 | 0.7 | 23.9 | 3.50 | 0.00 | 292522 | 0.6 | 5.0 | 3.68 | 0.00 | 285770 | 0.5 | 6.1 | 3.30 | 0.00 |
| 25 | LL | 238797 | 1.6 | 14.2 | 0.80 | 0.00 | 231983 | 1.4 | 13.2 | 1.17 | 0.00 | 226677 | 1.4 | 7.2 | 1.11 | 0.00 |
| 25 | LT | 278288 | 2.4 | 79.5 | 3.90 | 0.00 | 270489 | 1.1 | 43.8 | 4.68 | 0.00 | 262717 | 2.0 | 17.6 | 5.12 | 0.00 |
| 25 | TL | 310264 | 2.2 | 12.5 | 0.71 | 0.00 | 303368 | 1.1 | 6.1 | 1.06 | 0.00 | 297465 | 1.1 | 5.9 | 0.89 | 0.00 |
| 25 | TT | 350285 | 1.1 | 78.7 | 5.13 | 0.00 | 342485 | 2.0 | 103.7 | 4.87 | 0.00 | 334713 | 1.3 | 35.0 | 4.19 | 0.00 |
| 40 | LL | 241905 | 6.2 | 50.3 | 1.04 | 0.00 | 236333 | 6.8 | 72.7 | 1.42 | 0.00 | 231725 | 6.8 | 69.1 | 1.40 | 0.00 |
| 40 | LT | 273558 | 6.2 | 106.8 | 2.59 | 0.00 | 267193 | 11.2 | 72.6 | 3.28 | 0.00 | 261029 | 5.5 | 58.2 | 3.10 | 0.00 |
| 40 | TL | 298863 | 6.4 | 38.3 | 0.29 | 0.00 | 294942 | 6.0 | 35.1 | 0.58 | 0.00 | 290704 | 5.5 | 28.7 | 1.02 | 0.00 |
| 40 | TT | 356250 | 6.3 | 310.8 | 3.74 | 0.00 | 349318 | 6.0 | 213.3 | 4.48 | 0.00 | 341032 | 10.4 | 125.5 | 4.13 | 0.00 |
| 50 | LL | 238551 | 25.1 | 82.5 | 0.75 | 0.00 | 232482 | 24.9 | 72.8 | 1.37 | 0.00 | 227602 | 13.8 | 97.5 | 1.89 | 0.00 |
| 50 | LT | 277268 | 18.0 | 3324.4 | 3.93 | 0.00 | 271889 | 25.7 | 4021.8 | 5.49 | 0.00 | 264800 | 23.1 | 5136.9 | 5.49 | 0.00 |
| 50 | TL | 319352 | 25.9 | 282.7 | 2.87 | 0.00 | 314931 | 26.5 | 166.8 | 2.30 | 0.00 | 308085 | 25.3 | 89.4 | 1.39 | 0.00 |
| 50 | TT | 420285 | 30.1 | 7200.0 | 6.23 | 1.06 | 415001 | 27.0 | 7200.0 | 7.11 | 1.38 | 408205 | 24.6 | 7200.0 | 7.10 | 1.75 |
| 60 | LL | 225967 | 48.2 | 151.7 | 0.84 | 0.00 | 220090 | 27.8 | 156.3 | 1.26 | 0.00 | 215070 | 48.5 | 221.2 | 1.14 | 0.00 |
| 60 | LT | 257983 | 60.8 | 5020.0 | 3.73 | 0.00 | 253689 | 55.2 | 7200.0 | 5.99 | 1.61 | 246271 | 48.0 | 7200.0 | 5.88 | 0.30 |
| 60 | TL | 252437 | 45.3 | 236.0 | 0.82 | 0.00 | 246404 | 26.3 | 155.3 | 1.05 | 0.00 | 241525 | 25.8 | 220.9 | 0.97 | 0.00 |
| 60 | TT | 352480 | 31.6 | 401.1 | 2.75 | 0.00 | 345384 | 69.4 | 698.3 | 3.26 | 0.00 | 337734 | 52.5 | 235.3 | 3.02 | 0.00 |
| 70 | LL | 236810 | 68.6 | 2290.9 | 1.59 | 0.00 | 230513 | 48.2 | 1074.9 | 2.09 | 0.00 | 225372 | 48.9 | 1391.8 | 2.48 | 0.00 |
| 70 | LT | 258805 | 75.1 | 7200.0 | 3.39 | 1.01 | 258442 | 95.0 | 7200.0 | 6.87 | 3.36 | 250383 | 51.4 | 7200.0 | 6.68 | 3.10 |
| 70 | TL | 271170 | 48.0 | 438.1 | 1.12 | 0.00 | 261812 | 41.7 | 215.6 | 1.47 | 0.00 | 254343 | 80.6 | 424.7 | 1.42 | 0.00 |
| 70 | TT | 390172 | 57.5 | 441.5 | 4.12 | 0.00 | 384757 | 47.8 | 804.7 | 5.10 | 0.00 | 378168 | 53.4 | 406.5 | 5.06 | 0.00 |
| 75 | LL | 238171 | 77.1 | 371.0 | 0.61 | 0.00 | 233202 | 66.3 | 448.2 | 1.17 | 0.00 | 228904 | 58.3 | 713.8 | 1.35 | 0.00 |
| 75 | LT | 257357 | 155.4 | 4447.0 | 2.62 | 0.00 | 252688 | 71.7 | 7200.0 | 3.78 | 1.05 | 248018 | 59.3 | 7200.0 | 4.24 | 1.26 |
| 75 | TL | 303354 | 72.2 | 425.8 | 1.06 | 0.00 | 297263 | 65.6 | 368.8 | 2.06 | 0.00 | 292167 | 60.6 | 690.4 | 2.70 | 0.00 |
| 75 | TT | 351008 | 74.2 | 893.1 | 5.70 | 0.00 | 344801 | 63.6 | 1175.5 | 6.68 | 0.00 | 336945 | 57.5 | 2002.3 | 6.54 | 0.00 |
| 90 | LL | 224255 | 138.3 | 873.4 | 0.98 | 0.00 | 217859 | 114.0 | 1173.0 | 1.22 | 0.00 | 212624 | 114.7 | 1218.4 | 1.19 | 0.00 |
| 90 | LT | 246946 | 251.2 | 7200.0 | 2.09 | 0.57 | 258583 | 341.1 | 7200.0 | 9.32 | 7.87 | 242328 | 265.5 | 7200.0 | 5.43 | 2.92 |
| 90 | TL | 283353 | 149.4 | 1189.1 | 2.41 | 0.00 | 277535 | 125.6 | 1686.2 | 3.54 | 0.00 | 271391 | 223.6 | 1675.6 | 3.60 | 0.00 |
| 90 | TT | 340394 | 141.2 | 1020.4 | 2.58 | 0.00 | 334119 | 214.5 | 1981.7 | 3.76 | 0.00 | 326545 | 111.9 | 1633.6 | 3.64 | 0.00 |
| 100 | LL | 248282 | 561.3 | 7200.0 | 2.40 | 1.89 | 242773 | 247.0 | 7200.0 | 3.29 | 1.57 | 236676 | 196.4 | 7200.0 | 3.03 | 1.24 |
| 100 | LT | 257579 | 679.4 | 7200.0 | 2.27 | 0.47 | 251508 | 280.0 | 7200.0 | 3.03 | 0.65 | 245493 | 261.2 | 6586.7 | 2.86 | 0.00 |
| 100 | TL | 362765 | 398.8 | 2917.8 | 5.54 | 0.00 | 355930 | 343.3 | 2751.1 | 5.69 | 0.00 | 350611 | 320.8 | 1803.6 | 5.84 | 0.00 |
| 100 | TT | 480693 | 760.7 | 6538.4 | 3.96 | 0.00 | 472964 | 493.1 | 7200.0 | 5.45 | 0.63 | 461158 | 198.5 | 7200.0 | 5.24 | 0.58 |
| 125 | LL | 241875 | 860.0 | 7200.0 | 2.88 | 2.36 | 236946 | 587.6 | 7200.0 | 4.12 | 3.81 | 237294 | 482.9 | 7200.0 | 7.12 | 6.73 |
| 125 | LT | 254320 | 1678.8 | 7200.0 | 2.44 | 1.45 | 250810 | 604.3 | 7200.0 | 4.72 | 4.28 | 252902 | 833.8 | 7200.0 | 8.38 | 7.88 |
| 125 | TL | 246470 | 456.8 | 2443.0 | 0.98 | 0.00 | 240320 | 413.0 | 2183.2 | 1.23 | 0.00 | 235225 | 692.3 | 4472.7 | 1.09 | 0.00 |
| 125 | TT | 296995 | 487.1 | 2576.4 | 3.12 | 0.00 | 290641 | 777.7 | 2300.9 | 5.37 | 0.00 | 281396 | 423.8 | 2154.7 | 5.41 | 0.00 |
| 150 | LL | 302264 | 3309.3 | 7200.0 | 23.15 | 23.05 | 228462 | 1354.0 | 7200.0 | 1.93 | 1.64 | 223084 | 2080.4 | 7200.0 | 2.42 | 2.08 |
| 150 | LT | 259217 | 4346.1 | 7200.0 | 5.82 | 5.66 | 249953 | 1747.5 | 7200.0 | 5.60 | 5.22 | 243482 | 1192.9 | 7200.0 | 5.63 | 5.20 |
| 150 | TL | 262388 | 1206.7 | 7200.0 | 2.03 | 1.44 | 254937 | 2038.8 | 6479.0 | 2.42 | 0.00 | 249068 | 997.7 | 6662.9 | 2.53 | 0.00 |
| 150 | TT | 337471 | 1752.3 | 7200.0 | 9.07 | 7.44 | 333165 | 1604.3 | 7200.0 | 10.66 | 8.82 | 316975 | 1123.4 | 7200.0 | 8.11 | 6.25 |
| 175 | LL | N/A | 3000.9 | 7200.0 | N/A | N/A | N/A | 3367.4 | 7200.0 | N/A | N/A | N/A | 3201.6 | 7200.0 | N/A | N/A |
| 175 | LT | N/A | 5298.9 | 7200.0 | N/A | N/A | N/A | 4393.2 | 7200.0 | N/A | N/A | N/A | 3619.4 | 7200.0 | N/A | N/A |
| 175 | TL | N/A | 1846.7 | 7200.0 | N/A | N/A | N/A | 3250.1 | 7200.0 | N/A | N/A | N/A | 3023.5 | 7200.0 | N/A | N/A |
| 175 | TT | 355056 | 4611.6 | 7200.0 | 14.88 | 14.75 | N/A | 5989.5 | 7200.0 | N/A | N/A | 342868 | 3859.3 | 7200.0 | 15.15 | 14.85 |
| 200 | LL | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 4063.5 | 7200.0 | N/A | N/A | N/A | 6130.1 | 7200.0 | N/A | N/A |
| 200 | LT | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 7200.0 | 7200.0 | N/A | N/A |
| 200 | TL | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 5074.9 | 7200.0 | N/A | N/A | N/A | 3268.3 | 7200.0 | N/A | N/A |
| 200 | TT | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 7200.0 | 7200.0 | N/A | N/A | N/A | 6429.7 | 7200.0 | N/A | N/A |
| | | Average | 1101.5 | 3105.6 | 3.34 | 1.25 | | 940.1 | 3209.1 | 3.63 | 0.87 | | 839.7 | 3254.2 | 3.91 | 1.11 |
| | | Geometric mean | 64.0 | 519.3 | | | | 57.0 | 505.7 | | | | 52.2 | 497.9 | | |

N/A: Not available due to time limit

uncapacitated case. The solver can identify integer feasible solutions for instances up to 150 nodes, but there are unsolved instances even for 50 nodes. Moreover, the LP relaxation cannot be solved for 200 nodes.

### 5.3. Algorithmic performance

This section provides computational results which illustrate the performance of the proposed branch-and-cut algorithms BC-LS, BC-LR, and BC-LR+ for the solution of USAHLP-h and CSAHLP-h. In Table 4, we compare the three algorithms in terms of computational times. Each row of the table report the average results on six instances for the USAHLP-h and 12 instances for the CSAHLP-h. For each algorithm, the 'Time' column provides the average computational time and '#solved' column shows the total number of instances solved to optimality within the time limit.

For the USAHLP-h, the BC-LS algorithm, can solve only 25 of total 84 instances within an average of 5207.5 seconds. The performance of the BC-LR is much better than that of the BC-LS since it can solve instances with up to 200 nodes within an average solution time of 1550.6 seconds. The total number of instances solved to optimality using the BC-LR is 69 out of 84. The BC-LR+ algorithm, on the other hand, has the best performance; it solves all instances to optimality within an average time of 206.3 seconds. Of those 25 instances that could be solved to optimality by all BC-LS, BC-LR and BC-LR+ algorithms, the average solution times are 505.3, 5.63 and 0.46 seconds, respectively. The acceleration factor is three orders of magnitude.

The performance of the algorithm is similar for CSAHLP-h and USAHLP-h. The BC-LR+ is the best with average solution time of 822.9 seconds. Of the 40 instances that could be solved to optimality by all BC-LS, BC-LR and BC-LR+ algorithms, the average solution times are 624.2, 9.0, 0.5 seconds, respectively. The acceleration factor is again three orders of magnitude. BC-LR+ solves 156 out of 168 instances, which is 25% and 74% more than BC-LR and BC-LS, respectively. In Figure 2 we plot the behavior of the three algorithms on a sample CSAHLP-h instance of type $LT$ with $n = 50$ that can be solved by all algorithms within the time limit. The horizontal axis represent the time in seconds on a logarithmic scale, whereas the vertical axis shows the lower bound obtained by each algorithm. The figure shows how each algorithm converges to an optimal value as a function of the computing time. The BC-LR+ algorithm can reach the optimal value in a portion of the time needed by the other algorithms. The BC-LS algorithm has a poor performance, while BC-LR reaches an optimum within a reasonable time. Overall, the results indicate the superiority of BC-LR+.

**Table 4** Average solution time (seconds) of *BC-LS*, *BC-LR*, and *BC-LR+* on USAHLP-h and CSAHLP-h instances

| | USAHLP-h | | | | | | CSAHLP-h | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | *BC-LS* | | *BC-LR* | | *BC-LR+* | | *BC-LS* | | *BC-LR* | | *BC-LR+* | |
| Nodes | Time (s) | #solved | Time (s) | #solved | Time (s) | #solved | Time (s) | #solved | Time (s) | #solved | Time (s) | #solved |
| 10 | 1.7 | 6 | 0.6 | 6 | 0.0 | 6 | 3.1 | 12 | 0.6 | 12 | 0.0 | 12 |
| 20 | 29.3 | 6 | 1.8 | 6 | 0.4 | 6 | 181.5 | 12 | 3.0 | 12 | 0.4 | 12 |
| 25 | 135.8 | 6 | 8.4 | 6 | 0.2 | 6 | 936.4 | 12 | 13.8 | 12 | 0.5 | 12 |
| 40 | 4710.2 | 3 | 10.1 | 6 | 1.0 | 6 | 5332.1 | 3 | 35.6 | 12 | 2.4 | 12 |
| 50 | 7200.0 | 0 | 43.7 | 6 | 1.8 | 6 | 6907.3 | 1 | 176.6 | 12 | 21.2 | 12 |
| 60 | 7200.0 | 0 | 99.5 | 6 | 3.1 | 6 | 7200.0 | 0 | 297.2 | 12 | 23.6 | 12 |
| 70 | 3615.4 | 3 | 67.4 | 6 | 10.3 | 6 | 7200.0 | 0 | 721.2 | 12 | 30.2 | 12 |
| 75 | 6813.0 | 1 | 166.4 | 6 | 9.4 | 6 | 7200.0 | 0 | 1841.1 | 10 | 169.0 | 12 |
| 90 | 7200.0 | 0 | 390.5 | 6 | 32.7 | 6 | 7200.0 | 0 | 4384.5 | 5 | 306.0 | 12 |
| 100 | 7200.0 | 0 | 562.8 | 6 | 25.5 | 6 | 7200.0 | 0 | 3684.1 | 8 | 821.1 | 11 |
| 125 | 7200.0 | 0 | 3164.3 | 4 | 107.7 | 6 | 7200.0 | 0 | 4575.4 | 6 | 576.2 | 12 |
| 150 | 7200.0 | 0 | 6439.0 | 1 | 332.6 | 6 | 7200.0 | 0 | 6728.2 | 1 | 3796.7 | 7 |
| 175 | 7200.0 | 0 | 6064.1 | 1 | 376.4 | 6 | 7200.0 | 0 | 6431.5 | 2 | 2146.1 | 11 |
| 200 | 7200.0 | 0 | 4689.6 | 3 | 1986.7 | 6 | 7200.0 | 0 | 7081.1 | 1 | 3627.2 | 7 |
| Total | | **25/84** | | **69/84** | | **84/84** | | **40/168** | | **117/168** | | **156/168** |
| Average | 5207.5 | | 1550.6 | | 206.3 | | 5582.9 | | 2569.6 | | 822.9 | |
| Geometric mean | 1856.9 | | 132.7 | | 9.5 | | 2686.5 | | 383.3 | | 47.0 | |

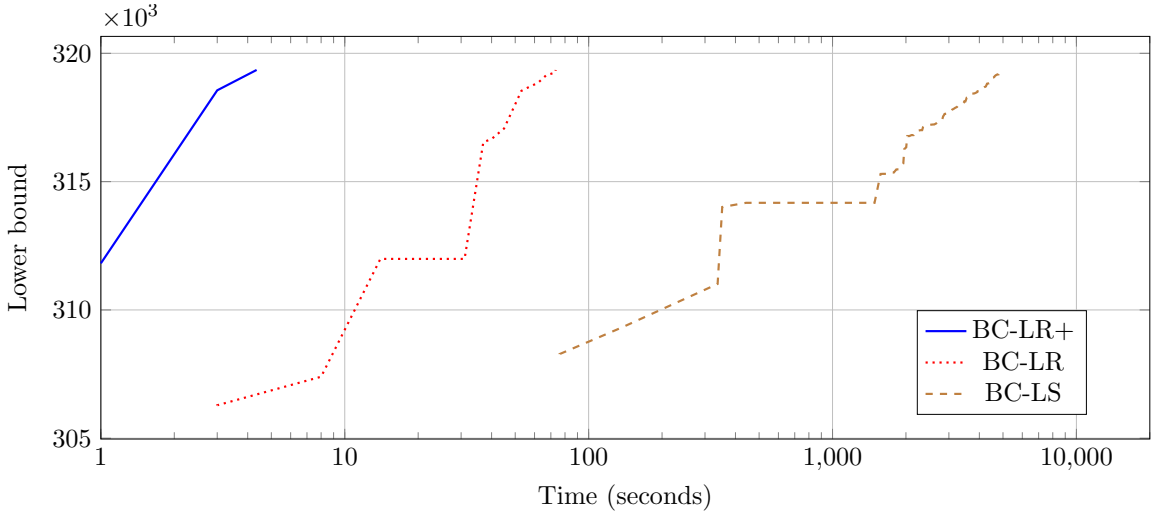


**Figure 2** Comparison of the BC-LS, BC-LR, and BC-LR+ on the sample CSAHLP-h instance of type *LT* from the AP I dataset with $n = 50$. The time axis is in logarithmic scale.

## 5.4. Detailed performance of the BC-LR+ algorithm

We now evaluate in more detail the performance of BC-LR+ in solving SAHLP-h. In Tables 5 and 6, we report the results for USAHLP-h and CSAHLP-h, respectively. The first two columns of these tables are the same as those of Tables 2 and 3. For every instance, we report the best upper bound found by the algorithm (UB), the optimality gap in percentage (Gap (%)), the solution time (Time), the number of nodes explored in the branch-and-cut tree (#bc), and the total number of Bender cuts generated in the branch-and-cut tree (#cuts).

For USAHLP-h in Table 5, the algorithm can solve all instances of each dataset to optimality with average computing times of 452.8, 117.9, and 48.1 for the AP I, AP II, and AP III, respectively. The AP I is the most difficult dataset, while AP III is the easiest. This is an expected behavior since the discount factors $\alpha$ are the largest in AP I and the smallest in AP III. As can be seen from Table 6, the algorithm has a similar performance on capacitated and uncapacitated instances. The AP I dataset is again the most difficult one, while the AP III is the easiest of the three datasets. The algorithm can solve instances with up to 200 nodes from each dataset. Instances with tight capacities (types LT and TT) are more difficult than those with larger capacities.

**Table 5** Performance of *BC-LR+* on the USAHLP-h instances

| | | AP I | | | | | AP II | | | | | AP III | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Nodes | Cost | UB | Gap (%) | Time (s) | #bc | cuts | UB | Gap (%) | Time (s) | #bc | cuts | UB | Gap (%) | Time (s) | #bc | cuts |
| 10 | L | 225928 | 0.00 | 0.0 | 37 | 21 | 220117 | 0.00 | 0.0 | 38 | 23 | 213652 | 0.00 | 0.0 | 5 | 4 |
| 10 | T | 264452 | 0.00 | 0.1 | 52 | 29 | 257614 | 0.00 | 0.0 | 39 | 20 | 251412 | 0.00 | 0.1 | 12 | 8 |
| 20 | L | 234585 | 0.00 | 0.5 | 21 | 13 | 228164 | 0.00 | 0.2 | 24 | 12 | 223111 | 0.00 | 0.1 | 15 | 7 |
| 20 | T | 271010 | 0.00 | 1.7 | 17 | 7 | 263845 | 0.00 | 0.1 | 19 | 9 | 258207 | 0.00 | 0.0 | 7 | 5 |
| 25 | L | 236650 | 0.00 | 0.5 | 62 | 34 | 230917 | 0.00 | 0.3 | 23 | 18 | 226155 | 0.00 | 0.1 | 13 | 11 |
| 25 | T | 295668 | 0.00 | 0.1 | 10 | 8 | 289172 | 0.00 | 0.2 | 11 | 9 | 283384 | 0.00 | 0.1 | 8 | 9 |
| 40 | L | 240934 | 0.00 | 1.7 | 174 | 41 | 234628 | 0.00 | 2.2 | 143 | 56 | 229500 | 0.00 | 0.9 | 77 | 23 |
| 40 | T | 293165 | 0.00 | 0.4 | 4 | 3 | 293165 | 0.00 | 0.2 | 4 | 3 | 290704 | 0.00 | 0.3 | 8 | 7 |
| 50 | L | 237478 | 0.00 | 3.9 | 104 | 28 | 231825 | 0.00 | 2.7 | 131 | 36 | 227204 | 0.00 | 1.8 | 81 | 24 |
| 50 | T | 300421 | 0.00 | 1.4 | 30 | 10 | 294551 | 0.00 | 0.3 | 3 | 3 | 288141 | 0.00 | 0.4 | 3 | 4 |
| 60 | L | 228885 | 0.00 | 4.3 | 142 | 39 | 222745 | 0.00 | 1.5 | 15 | 12 | 217723 | 0.00 | 0.9 | 7 | 6 |
| 60 | T | 264628 | 0.00 | 9.3 | 106 | 44 | 260273 | 0.00 | 1.7 | 23 | 15 | 255780 | 0.00 | 1.1 | 5 | 4 |
| 70 | L | 226181 | 0.00 | 41.9 | 445 | 115 | 220060 | 0.00 | 13.6 | 121 | 40 | 215019 | 0.00 | 4.6 | 24 | 12 |
| 70 | T | 261295 | 0.00 | 0.5 | 0 | 1 | 261295 | 0.00 | 0.5 | 0 | 1 | 261295 | 0.00 | 0.5 | 0 | 1 |
| 75 | L | 235756 | 0.00 | 37.5 | 290 | 84 | 229630 | 0.00 | 4.7 | 68 | 16 | 224719 | 0.00 | 3.0 | 19 | 12 |
| 75 | T | 288778 | 0.00 | 3.7 | 7 | 7 | 287687 | 0.00 | 3.0 | 5 | 6 | 284805 | 0.00 | 4.7 | 9 | 10 |
| 90 | L | 225512 | 0.00 | 77.4 | 1081 | 175 | 219494 | 0.00 | 51.4 | 415 | 115 | 214440 | 0.00 | 19.9 | 96 | 50 |
| 90 | T | 257477 | 0.00 | 22.3 | 216 | 59 | 252975 | 0.00 | 17.0 | 126 | 47 | 249242 | 0.00 | 8.4 | 16 | 11 |
| 100 | L | 238097 | 0.00 | 91.5 | 328 | 73 | 232477 | 0.00 | 26.8 | 188 | 37 | 227857 | 0.00 | 28.3 | 113 | 46 |
| 100 | T | 305098 | 0.00 | 2.4 | 0 | 2 | 305098 | 0.00 | 1.3 | 0 | 2 | 302775 | 0.00 | 2.4 | 4 | 4 |
| 125 | L | 227969 | 0.00 | 359.0 | 850 | 157 | 221732 | 0.00 | 116.8 | 412 | 86 | 216589 | 0.00 | 73.7 | 176 | 60 |
| 125 | T | 259065 | 0.00 | 67.0 | 231 | 61 | 252500 | 0.00 | 12.2 | 12 | 11 | 246993 | 0.00 | 17.6 | 9 | 8 |
| 150 | L | 225455 | 0.00 | 941.7 | 3395 | 405 | 218728 | 0.00 | 138.6 | 218 | 59 | 214684 | 0.00 | 136.3 | 203 | 62 |
| 150 | T | 234895 | 0.00 | 704.0 | 3379 | 353 | 228002 | 0.00 | 24.1 | 17 | 6 | 222629 | 0.00 | 50.5 | 41 | 26 |
| 175 | L | 227758 | 0.00 | 1469.8 | 3346 | 392 | 221122 | 0.00 | 272.1 | 344 | 69 | 215683 | 0.00 | 153.4 | 155 | 42 |
| 175 | T | 247539 | 0.00 | 160.1 | 466 | 51 | 240686 | 0.00 | 20.0 | 4 | 5 | 235928 | 0.00 | 183.1 | 36 | 28 |
| 200 | L | 233959 | 0.00 | 6548.6 | 8916 | 932 | 227880 | 0.00 | 2092.1 | 633 | 123 | 222838 | 0.00 | 196.9 | 101 | 32 |
| 200 | T | 272445 | 0.00 | 2128.4 | 3591 | 356 | 266077 | 0.00 | 497.3 | 432 | 85 | 259262 | 0.00 | 457.0 | 248 | 73 |
| | Average | | 0.00 | 452.8 | 975 | 125 | | 0.00 | 117.9 | 124 | 33 | | 0.00 | 48.1 | 53 | 21 |
| | Geometric mean | | | 11.8 | | | | | 4.0 | | | | | 3.1 | | |

## 6. Conclusions

We have studied a practical generalization of the classical hub location problem with flow-based discount factors for transportation costs, which we refer to as the single allocation hub location problem with heterogeneous economies of scale (SAHLP-h). We have developed a non-linear programming model which was reformulated as a MILP, and also a MIQCP with particular structures that can be exploited to develop Benders type decomposition methods with integer subproblems. Our models are independent of parameters and data specifications such as the triangle inequality.

**Table 6** Performance of *BC-LR+* on the CSAHLP-h instances

| | Cost/ | AP I | | | | | AP II | | | | | AP III | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Nodes | Capacity | UB | Gap (%) | Time (s) | #bc | cuts | UB | Gap (%) | Time (s) | #bc | cuts | UB | Gap (%) | Time (s) | #bc | cuts |
| 10 | LL | 225928 | 0.00 | 0.2 | 33 | 13 | 220117 | 0.00 | 0.0 | 38 | 23 | 213652 | 0.00 | 0.0 | 5 | 4 |
| 10 | LT | 255755 | 0.00 | 0.1 | 54 | 29 | 248992 | 0.00 | 0.0 | 62 | 34 | 241356 | 0.00 | 0.0 | 43 | 29 |
| 10 | TL | 264452 | 0.00 | 0.0 | 31 | 12 | 257614 | 0.00 | 0.0 | 36 | 20 | 251412 | 0.00 | 0.0 | 12 | 8 |
| 10 | TT | 264452 | 0.00 | 0.0 | 19 | 8 | 257614 | 0.00 | 0.0 | 18 | 11 | 252218 | 0.00 | 0.0 | 13 | 7 |
| 20 | LL | 234585 | 0.00 | 0.1 | 17 | 7 | 228164 | 0.00 | 0.1 | 14 | 5 | 223111 | 0.00 | 0.1 | 24 | 15 |
| 20 | LT | 253352 | 0.00 | 2.6 | 194 | 29 | 249067 | 0.00 | 0.3 | 152 | 24 | 245480 | 0.00 | 0.2 | 158 | 40 |
| 20 | TL | 271010 | 0.00 | 1.0 | 17 | 6 | 263845 | 0.00 | 0.1 | 7 | 3 | 258207 | 0.00 | 0.1 | 7 | 5 |
| 20 | TT | 298498 | 0.00 | 0.1 | 112 | 19 | 292522 | 0.00 | 0.3 | 75 | 9 | 285770 | 0.00 | 0.1 | 62 | 10 |
| 25 | LL | 238797 | 0.00 | 0.8 | 203 | 32 | 231983 | 0.00 | 0.7 | 166 | 24 | 226677 | 0.00 | 0.2 | 74 | 20 |
| 25 | LT | 278288 | 0.00 | 0.9 | 550 | 50 | 270489 | 0.00 | 0.5 | 253 | 36 | 262717 | 0.00 | 0.9 | 254 | 31 |
| 25 | TL | 310264 | 0.00 | 0.2 | 30 | 18 | 303368 | 0.00 | 0.2 | 19 | 12 | 297465 | 0.00 | 0.2 | 3 | 4 |
| 25 | TT | 350285 | 0.00 | 1.1 | 654 | 41 | 342485 | 0.00 | 0.6 | 440 | 34 | 334713 | 0.00 | 0.1 | 53 | 6 |
| 40 | LL | 241905 | 0.00 | 2.0 | 101 | 27 | 236333 | 0.00 | 2.3 | 85 | 32 | 231725 | 0.00 | 1.4 | 82 | 40 |
| 40 | LT | 273558 | 0.00 | 2.9 | 785 | 35 | 267193 | 0.00 | 2.6 | 671 | 28 | 261029 | 0.00 | 1.7 | 578 | 18 |
| 40 | TL | 298863 | 0.00 | 1.2 | 88 | 14 | 294942 | 0.00 | 0.4 | 32 | 10 | 290704 | 0.00 | 0.3 | 7 | 4 |
| 40 | TT | 356250 | 0.00 | 6.4 | 3079 | 83 | 349318 | 0.00 | 5.4 | 1163 | 46 | 341032 | 0.00 | 2.0 | 872 | 32 |
| 50 | LL | 238551 | 0.00 | 4.6 | 118 | 36 | 232482 | 0.00 | 2.8 | 74 | 21 | 227602 | 0.00 | 1.3 | 32 | 18 |
| 50 | LT | 277268 | 0.00 | 45.7 | 3322 | 151 | 271889 | 0.00 | 12.4 | 1802 | 51 | 264800 | 0.00 | 4.7 | 754 | 30 |
| 50 | TL | 319352 | 0.00 | 4.4 | 138 | 30 | 314931 | 0.00 | 4.1 | 204 | 51 | 308085 | 0.00 | 0.6 | 9 | 5 |
| 50 | TT | 419435 | 0.00 | 129.6 | 19120 | 233 | 414928 | 0.00 | 25.3 | 8210 | 117 | 407698 | 0.00 | 18.4 | 4345 | 77 |
| 60 | LL | 225967 | 0.00 | 9.7 | 346 | 74 | 220090 | 0.00 | 6.0 | 189 | 47 | 215070 | 0.00 | 3.3 | 65 | 25 |
| 60 | LT | 257983 | 0.00 | 56.3 | 6117 | 153 | 253403 | 0.00 | 106.2 | 4772 | 171 | 246245 | 0.00 | 15.2 | 1291 | 43 |
| 60 | TL | 252437 | 0.00 | 7.1 | 244 | 67 | 246404 | 0.00 | 2.3 | 33 | 19 | 241525 | 0.00 | 1.2 | 8 | 9 |
| 60 | TT | 352480 | 0.00 | 44.5 | 7759 | 194 | 345384 | 0.00 | 26.5 | 2287 | 76 | 337734 | 0.00 | 4.8 | 386 | 30 |
| 70 | LL | 236810 | 0.00 | 17.5 | 394 | 68 | 230513 | 0.00 | 19.5 | 274 | 40 | 225372 | 0.00 | 23.8 | 348 | 59 |
| 70 | LT | 258805 | 0.00 | 69.3 | 4013 | 104 | 253255 | 0.00 | 67.5 | 2822 | 110 | 247735 | 0.00 | 38.2 | 1156 | 34 |
| 70 | TL | 271170 | 0.00 | 15.3 | 478 | 78 | 261812 | 0.00 | 4.6 | 46 | 27 | 254343 | 0.00 | 2.2 | 18 | 12 |
| 70 | TT | 390172 | 0.00 | 73.2 | 9236 | 308 | 384757 | 0.00 | 24.8 | 1527 | 103 | 378168 | 0.00 | 6.3 | 172 | 10 |
| 75 | LL | 238171 | 0.00 | 11.6 | 217 | 42 | 233202 | 0.00 | 19.2 | 228 | 36 | 228904 | 0.00 | 16.0 | 189 | 32 |
| 75 | LT | 257357 | 0.00 | 24.4 | 1435 | 43 | 252688 | 0.00 | 41.5 | 1837 | 59 | 248007 | 0.00 | 31.6 | 884 | 27 |
| 75 | TL | 303354 | 0.00 | 22.4 | 813 | 91 | 297263 | 0.00 | 10.5 | 427 | 46 | 292167 | 0.00 | 4.6 | 319 | 19 |
| 75 | TT | 351008 | 0.00 | 1749.1 | 780711 | 467 | 344801 | 0.00 | 56.2 | 21445 | 133 | 336945 | 0.00 | 41.2 | 3415 | 72 |
| 90 | LL | 224255 | 0.00 | 24.1 | 243 | 60 | 217859 | 0.00 | 10.9 | 60 | 27 | 212624 | 0.00 | 4.6 | 5 | 5 |
| 90 | LT | 246946 | 0.00 | 1184.1 | 38822 | 507 | 242138 | 0.00 | 153.6 | 8400 | 144 | 237567 | 0.00 | 89.5 | 1141 | 47 |
| 90 | TL | 283353 | 0.00 | 145.2 | 19669 | 268 | 277535 | 0.00 | 45.8 | 4704 | 99 | 271391 | 0.00 | 13.7 | 925 | 24 |
| 90 | TT | 340394 | 0.00 | 1588.7 | 258808 | 792 | 334119 | 0.00 | 186.6 | 37768 | 228 | 326545 | 0.00 | 225.5 | 15298 | 82 |
| 100 | LL | 248282 | 0.00 | 209.4 | 1548 | 90 | 242773 | 0.00 | 92.6 | 1231 | 63 | 236676 | 0.00 | 43.6 | 612 | 22 |
| 100 | LT | 257579 | 0.00 | 389.9 | 7666 | 237 | 251508 | 0.00 | 149.8 | 2022 | 67 | 245493 | 0.00 | 50.5 | 744 | 41 |
| 100 | TL | 362765 | 0.00 | 183.5 | 4179 | 145 | 355930 | 0.00 | 50.7 | 1004 | 70 | 350611 | 0.00 | 29.1 | 538 | 38 |
| 100 | TT | 480693 | 0.22 | 7200.0 | 255115 | 1730 | 472964 | 0.00 | 1330.1 | 119683 | 609 | 461158 | 0.00 | 124.4 | 16728 | 131 |
| 125 | LL | 240450 | 0.00 | 416.3 | 2473 | 81 | 234728 | 0.00 | 457.5 | 3030 | 114 | 228022 | 0.00 | 145.9 | 956 | 52 |
| 125 | LT | 254320 | 0.00 | 2234.8 | 13646 | 469 | 250536 | 0.00 | 1078.4 | 5326 | 177 | 244991 | 0.00 | 181.3 | 1478 | 78 |
| 125 | TL | 246470 | 0.00 | 1041.0 | 3373 | 522 | 240320 | 0.00 | 114.8 | 355 | 107 | 235225 | 0.00 | 31.1 | 46 | 30 |
| 125 | TT | 296995 | 0.00 | 1043.3 | 16947 | 455 | 290641 | 0.00 | 150.2 | 3164 | 129 | 281396 | 0.00 | 19.2 | 467 | 9 |
| 150 | LL | 234844 | 0.00 | 1182.6 | 1476 | 214 | 228462 | 0.00 | 1543.5 | 1396 | 211 | 223084 | 0.00 | 225.0 | 481 | 75 |
| 150 | LT | 253905 | 1.59 | 7200.0 | 215471 | 174 | 249517 | 1.69 | 7200.0 | 302562 | 60 | 243482 | 1.54 | 7200.0 | 665118 | 30 |
| 150 | TL | 262388 | 0.00 | 1368.6 | 16129 | 588 | 254937 | 0.00 | 697.4 | 3323 | 161 | 249068 | 0.00 | 236.2 | 814 | 50 |
| 150 | TT | 327474 | 0.32 | 7200.0 | 77388 | 1109 | 323591 | 0.08 | 7200.0 | 160513 | 1589 | 315767 | 0.00 | 4307.0 | 153331 | 766 |
| 175 | LL | 228001 | 0.00 | 1716.9 | 7887 | 425 | 222226 | 0.00 | 902.0 | 2619 | 203 | 215802 | 0.00 | 125.5 | 509 | 25 |
| 175 | LT | 255214 | 0.00 | 3971.5 | 19475 | 337 | 250853 | 0.00 | 1432.9 | 9299 | 100 | 244415 | 0.00 | 2178.8 | 7412 | 118 |
| 175 | TL | 245016 | 0.00 | 594.8 | 1203 | 190 | 238548 | 0.00 | 196.3 | 237 | 57 | 232815 | 0.00 | 19.7 | 3 | 2 |
| 175 | TT | 315857 | 0.31 | 7200.0 | 172475 | 1096 | 311569 | 0.00 | 7200.0 | 128133 | 407 | 305466 | 0.00 | 354.5 | 12710 | 55 |
| 200 | LL | 232138 | 0.00 | 2578.3 | 5019 | 156 | 226507 | 0.00 | 921.2 | 2518 | 76 | 220363 | 0.00 | 716.0 | 1281 | 64 |
| 200 | LT | 272378 | 3.57 | 7200.0 | 20572 | 165 | 267396 | 4.76 | 7200.0 | 9779 | 118 | 259966 | 2.38 | 7200.0 | 17563 | 161 |
| 200 | TL | 274160 | 0.00 | 1148.2 | 1033 | 88 | 267006 | 0.00 | 427.0 | 967 | 64 | 259013 | 0.00 | 527.8 | 761 | 37 |
| 200 | TT | 295273 | 0.51 | 7200.0 | 119007 | 748 | 289606 | 0.17 | 7200.0 | 131982 | 736 | 281331 | 0.00 | 1207.6 | 22876 | 149 |
| | Average | | 0.00 | 1188.0 | 37858 | 236 | | 0.00 | 828.3 | 17669 | 126 | | 0.00 | 455.0 | 16740 | 51 |
| | Geometric mean | | | 44.6 | | | | | 21.6 | | | | | 9.5 | | |

We have investigated an application of the integer L-shaped decomposition algorithm to solve the MILP formulation, and we have developed a generalized Benders decomposition for the MIQCP. The latter is based on a Lagrangian relaxation, which yields a subproblem decomposition and a tight lower bound. We have used linear dual functions to underestimate the Lagrangian function in order to generate optimality cuts. We have also developed an efficient algorithm to generate enhanced cuts. We have performed extensive computational experiments on both uncapacitated and capacitated SAHLP-h instances. Our results indicate the efficacy of our algorithms on large-

scale instances derived from the classical Australian Post dataset. Our best algorithm was capable of solving instances with up to 200 nodes within one hour of computing time.

## Acknowledgments

We are grateful to two anonymous referees for their constructive comments and suggestions which helped improve the quality of the paper. The first author has been supported by Canadian Natural Sciences and Engineering Research Council (NSERC) Discovery Grant RGPIN- 2020-05395. The third and the fourth authors gratefully acknowledge funding provided by NSERC under grant 2015-06189. The computations reported in this paper were performed on Compute Canada.

## References

Alumur, S. A., Campbell, J. F., Contreras, I., Kara, B. Y., Marianov, V., and O'Kelly, M. E. (2020). Perspectives on modeling hub location problems. *European Journal of Operational Research*.

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.

Angulo, G., Ahmed, S., and Dey, S. S. (2016). Improving the integer L-shaped method. *INFORMS Journal on Computing*, 28(3):483–499.

Campbell, J. F. (1994). Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, 72(2):387–405.

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. (2005). Hub arc location problems: part I—introduction 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.

Carello, G., Della Croce, F., Ghirardi, M., and Tadei, R. (2004). Solving the hub location problem in telecommunication network design: A local search approach. *Networks*, 44(2):94–105.

Çetiner, S., Sepil, C., and Süral, H. (2010). Hubbing and routing in postal delivery systems. *Annals of Operations Research*, 181(1):109–124.

Contreras, I. (2015). Hub location problems. In Laporte, G., Nickel, S., and Saldanha da Gama, F., editors, *Location Science*, pages 311–344. Springer International Publishing, Cham.

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., Cordeau, J.-F., and Laporte, G. (2012). Exact solution of large-scale hub location problems with multiple capacity levels. *Transportation Science*, 46(4):439–459.

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. (2010a). The capacitated single-allocation hub location problem revisited: A note on a classical formulation. *European Journal of Operational Research*, 207(1):92–96.

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

Croxton, K. L., Gendron, B., and Magnanti, T. L. (2003a). A comparison of mixed-integer programming models for nonconvex piecewise linear cost minimization problems. *Management Science*, 49(9):1268–1273.

Croxton, K. L., Gendron, B., and Magnanti, T. L. (2003b). Models and methods for merge-in-transit operations. *Transportation Science*, 37(1):1–22.

Croxton, K. L., Gendron, B., and Magnanti, T. L. (2007). Variable disaggregation in network flow problems with piecewise linear costs. *Operations Research*, 55(1):146–157.

Cunha, C. B. and Silva, M. R. (2007). A genetic algorithm for the problem of configuring a hub-and-spoke network for a ltl trucking company in brazil. *European Journal of Operational Research*, 179(3):747–758.

de Camargo, R. S., de Miranda Jr, G., and Luna, H. P. L. (2009). Benders decomposition for hub location problems with economies of scale. *Transportation Science*, 43(1):86–97.

Dukkanci, O. and Kara, B. Y. (2017). Routing and scheduling decisions in the hierarchical hub location problem. *Computers & Operations Research*, 85:45–57.

Eiselt, H. A. and Marianov, V. (2009). A conditional $p$-hub location problem with attraction functions. *Computers & Operations Research*, 36(12):3128–3135.

Elhedhli, S. and Hu, F. X. (2005). Hub-and-spoke network design with congestion. *Computers & Operations Research*, 32(6):1615–1632.

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.

Fisher, M. L. (2004). The Lagrangian relaxation method for solving integer programming problems. *Management Science*, 50(12_supplement):1861–1871.

Frangioni, A. and Gendron, B. (2009). 0–1 reformulations of the multicommodity capacitated network design problem. *Discrete Applied Mathematics*, 157(6):1229–1241.

Gelareh, S. and Pisinger, D. (2011). Fleet deployment, network design and hub location of liner shipping companies. *Transportation Research Part E: Logistics and Transportation Review*, 47(6):947–964.

Guzelsoy, M. and Ralphs, T. K. (2007). Duality for mixed-integer linear programs. *International Journal of Operations Research*, 4(3):118–137.

Imai, A., Shintani, K., and Papadimitriou, S. (2009). Multi-port vs. hub-and-spoke port calls by containerships. *Transportation Research Part E: Logistics and Transportation Review*, 45(5):740–757.

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

Kara, B. Y. and Tansel, B. C. (2001). The latest arrival hub location problem. *Management Science*, 47(10):1408–1420.

Kelley, J. E. (1960). The cutting-plane method for solving convex programs. *Journal of the Society for Industrial and Applied Mathematics*, 8(4):703–712.

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

Klincewicz, J. G. (2002). Enumeration and search procedures for a hub location problem with economies of scale. *Annals of Operations Research*, 110(1-4):107–122.

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.

Magnanti, T. L. and Wong, R. T. (1981). Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, 29(3):464–484.

Mahéo, A., Kilby, P., and Van Hentenryck, P. (2019). Benders decomposition for the design of a hub and shuttle public transit system. *Transportation Science*, 53(1):77–88.

Meier, J. F. and Clausen, U. (2018). Solving single allocation hub location problems on Euclidean data. *Transportation Science*, 52(5):1141–1155.

Nickel, S., Schöbel, A., and Sonneborn, T. (2001). Hub location problems in urban traffic networks. In Pursula, M. and Niittymäki, J., editors, *Mathematical Methods on Optimization in Transportation Systems*, pages 95–107. Springer, Boston.

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.

O'Kelly, M. E. and Bryan, D. (1998). Hub location with flow economies of scale. *Transportation Research Part B: Methodological*, 32(8):605–616.

Podnar, H., Skorin-Kapov, J., and Skorin-Kapov, D. (2002). Network cost minimization using threshold-based discounting. *European Journal of Operational Research*, 137(2):371–386.

Ralphs, T. K. and Hassanzadeh, A. (2014). A generalization of Benders' algorithm for two-stage stochastic optimization problems with mixed integer recourse. Technical report, Technical Report 14T-005, Department of Industrial and Systems Engineering, Lehigh University, USA.

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.

Rostami, B., Kämmerling, N., Naoum-Sawaya, J., Buchheim, C., and Clausen, U. (2020). Stochastic single-allocation hub location. *European Journal of Operational Research*.

Rostami, B., Meier, J., Buchheim, C., and Clausen, U. (2015). The uncapacitated single allocation p-hub median problem with stepwise cost function. *Optimization Online*.

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.

Tanash, M., Contreras, I., and Vidyarthi, N. (2017). An exact algorithm for the modular hub location problem with single assignments. *Computers & Operations Research*, 85:32–44.

Vielma, J. P., Ahmed, S., and Nemhauser, G. (2010). Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Operations Research*, 58(2):303–315.

Wolsey, L. A. (1981). Integer programming duality: Price functions and sensitivity analysis. *Mathematical Programming*, 20(1):173–195.

Yaman, H., Karasan, O. E., and Kara, B. Y. (2012). Release time scheduling and hub location for next-day delivery. *Operations Research*, 60(4):906–917.