Hindawi
Journal of Advanced Transportation
Volume 2017, Article ID 9387302, 23 pages
https://doi.org/10.1155/2017/9387302

Review Article

Finding p-Hub Median Locations: An Empirical Study on Problems and Solution Techniques

Xiaoqian Sun,[1,2] Weibin Dai,[1,2] Yu Zhang,[3,4,5] and Sebastian Wandelt[1,2]

[1] *School of Electronic and Information Engineering, Beihang University, Beijing 100191, China*
[2] *Beijing Key Laboratory for Network-Based Cooperative ATM, Beijing 100191, China*
[3] *Department of Civil and Environmental Engineering, University of South Florida, Tampa, FL 33620, USA*
[4] *College of Air Traffic Management, Civil Aviation University of China, Tianjin 300300, China*
[5] *College of Transportation Engineering, Tongji University, Shanghai 200092, China*

Correspondence should be addressed to Sebastian Wandelt; wandelt@informatik.hu-berlin.de

Received 18 May 2017; Revised 22 October 2017; Accepted 31 October 2017; Published 23 November 2017

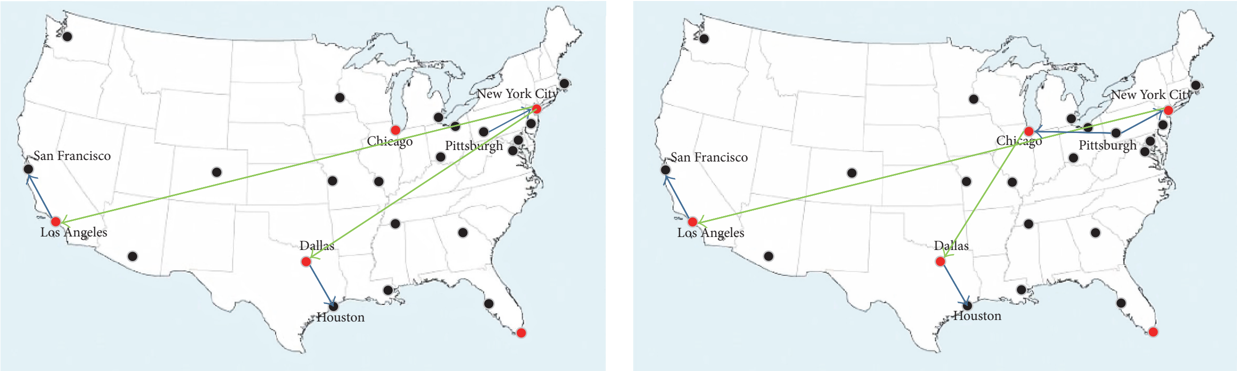
Academic Editor: Zhi-Chun Li

Hub location problems have been studied by many researchers for almost 30 years, and, accordingly, various solution methods have been proposed. In this paper, we implement and evaluate several widely used methods for solving five standard hub location problems. To assess the scalability and solution qualities of these methods, three well-known datasets are used as case studies: Turkish Postal System, Australia Post, and Civil Aeronautics Board. Classical problems in small networks can be solved efficiently using CPLEX because of their low complexity. Genetic algorithms perform well for solving three types of single allocation problems, since the problem formulations can be neatly encoded with chromosomes of reasonable size. Lagrangian relaxation is the only technique that solves reliable multiple allocation problems in large networks. We believe that our work helps other researchers to get an overview on the best solution techniques for the problems investigated in our study and also stipulates further interest on cross-comparing solution techniques for more expressive problem formulations.

1. Introduction

Hub location problems deal with the location of hub facilities in a network [1]. These problems are highly relevant in several fields, such as transportation [2–6] and telecommunication [7, 8]; see [9–11] for some excellent surveys. In general, flows are routed from origin nodes to destination nodes through a "spoke-hub-hub-spoke" structure. First, flows from each origin node are collected by their hubs; then, the flows are transferred through at most another hub; and finally, flows are distributed to their destinations. Because of the economies of scale, there are cost discounts while transporting flows between hubs; the total cost with the "spoke-hub-hub-spoke" structure may be less than that for transporting flows from origins to destinations directly [10].

If the number of hubs p is fixed at design time, the hub location problem is called p-hub median problem (pHMP) [12]. In this paper, we consider hub location problems where the number of hubs p is fixed. Based on the constraints for node assignment, hub location problems have two basic structures: single allocation (SA) and multiple allocation (MA) [13]. In SA problems, each node is allocated to a specific hub, and all inbound or outbound flows of each node must travel through its hub. In MA problems, on the other hand, the flows for different origin-destination (OD) pairs can be routed through different hubs. An example for the SA and MA problem is shown in Figure 1. Over time, many variations of the core problem, including uncertainty and reliability, have been considered by researchers. In early research, it is assumed that all nodes can always function properly. However, in reality, each hub can be disrupted because of possible accidents. For instance, in air transportation networks, nodes (i.e., airports) can be disrupted because of bad weather or equipment outage [14, 15]. Therefore, the reliability of nodes should be considered for more realistic modeling [11, 16], taking into account the actual topology [17]. In early research, the hub networks are assumed to be complete; that is, hubs are fully connected with each other. In recent years, incomplete



(a) Single allocation problem

(b) Multiple allocation problem

Figure 1: Single allocation problem and multiple allocation problem: (a) Flights from Pittsburgh must go through its hub New York City first for different destinations in single allocation problems. (b) The flow can be assigned to New York City for destination of San Francisco and to Chicago for destination of Houston in multiple allocation problems.

hub networks have also been studied [18]. In this case, some hubs are not directly connected.

Apart from a few exceptions, hub location problems are NP-hard [19, 20]. Therefore, many researchers have focused on proposing reasonable models and efficient algorithms for solving hub location problems. Reference [1] published the first paper about hub location problems, and the first mathematical formulation for pHMP in the following year was proposed [12]. After that, the pHMP has been studied by many researchers. Various types of efficient models and algorithms have been proposed. In early studies for hub location problems, researchers applied different solution methods for solving them. For instance, [21] proposed a heuristic method based on tabu search (TS) for solving single allocation problems. Reference [22] proposed a new linear formulation for single allocation model with fewer variables and constraints than the classical formulations. A heuristic based on simulated annealing algorithm (SAA) and a linear program based on branch-and-bound (BB) method are presented for solving the problem as well. After that, branch-and-bound methods and Lagrangian relaxation (LR) algorithm started to be applied to hub location problems. Reference [23] proposed a sophisticated approach based on Lagrangian relaxation for solving single allocation hub problems. Compared with simulated annealing and branch-and-bound method, the problem with 25 nodes can be solved within a reasonable time and solutions with high quality can be obtained. Reference [24] presented a branch-and-price algorithm (BP) for solving a capacitated single allocation problem. To obtain a tight lower bound, Lagrangian relaxation was applied to the problem. Instances with up to 200 nodes were solved optimally with their algorithm. Reference [25] studied the combination of congestion and capacity decisions in hub-and-spoke network design. They decomposed the problem into subproblems with Lagrangian relaxation. Reference [26] proposed new models with the consideration of uncertainties (hub failures and backup hubs) for single and multiple allocation problems. Based on the heuristics proposed by [23], they used Lagrangian relaxation and branch-and-bound method to solve reliable models for these two allocation problems.

In addition to Lagrangian relaxation and branch-and-bound method, genetic algorithms (GA) are also commonly used for solving hub location problems. Authors of [27] proposed an effective method based on a genetic search algorithm. They showed that the GA-based method outperforms other approaches in their study on both computation time and solution quality. The method has a potential for solving large problem instances. Based on their research, [28] proposed two genetic algorithms for solving an uncapacitated single allocation p-hub median problem. They compared their methods with a tabu search, simulated annealing, and path relinking method (PRM) [35]. Reference [30] proposed a genetic algorithm approach for solving a capacitated single allocation p-hub median problem, which can guarantee the optimality for networks with up to 50 nodes. The fuzzy capacitated p-hub center problem was considered by [36]. They provided a genetic algorithm method to solve it. Reference [37] proposed an efficient genetic algorithm for solving the liner hub-and-spoke shipping network design problem. The authors of [31] presented the formulation and a GA-based algorithm for solving a planar hub location problem in reasonable time. The Weiszfeld algorithm (WA) [38] and the location-allocation algorithm (LAA) [39] were also compared with their heuristic on different datasets. In addition, genetic algorithms were also used to solve the reliable hub location problem by [33]. With the consideration of hub disruptions, they proposed a mathematical model for single allocation cases. The proposed method performs better than CPLEX.

Based on traditional models, several variants of the problems with new constraints were also studied, such as reliability. Reference [8] presented new reliable models for maximizing expected flow between city nodes. Reference [40] proposed a compact mixed integer program model and a continuum approximation model for the reliable uncapacitated hub location problem. With a custom-design Lagrangian relaxation method, near-optimal solutions can be obtained. In addition to reliability, the incomplete hub networks have also been studied. Reference [18] proposed models for several single allocation problems under

Table 1: Summary of different allocation problems in the literature.

	Classical single	Classical multiple	Reliable single	Reliable multiple	Incomplete single
[12]	√				
[21]	√				
[22]	√				
[23]	√				
[27]	√				
[28]	√				
[18]					√
[8]			√	√	
[29]		√			
[24]	√				
[30]	√				
[26]			√	√	
[31]	√				
[32]	√				
[33]			√		
[34]					√

incomplete hub network design and the mathematical formulations with $O(n^3)$ variables were presented. Reference [34] also modeled the incomplete hub location problem with and without hop-constraints. Benders decomposition technique was used to solve the problem in large scales.

In recent years, new algorithms and applications on hub location problems have been proposed. Reference [41] proposed approximation algorithms for solving the single allocation problem. Reference [42] studied single allocation problems with multiple capacity levels, where the decision making process includes the size of hubs. An enhanced Benders decomposition (EBD) algorithm was used to solve multiple allocation uncapacitated hub location problem for large-scale networks by [29]. The algorithm was evaluated in computational experiments on different instances with up to 500 nodes, compared with Benders decomposition [43], adjustment procedure [44], relax-and-cut algorithm (RCA) [45], and CPLEX. Reference [46] proposed two formulations for capacitated ordered median hub location problems. Reference [32] proposed a new method for single allocation hub location analysis. Based on the spatial and flow properties of nodes, a clustering-based potential hub set (CBS) can be generated, which can help narrow the solution sets and reduce the computational complexity of the problems. In addition to the standard hub location problems above, several variants of hub location problems have also been studied. Reference [47] developed a model with equilibrium constraints for the intermodal hub-and-spoke problem and proposed a hybrid genetic algorithm for solving it. Reference [48] studied hub location problems under uncertainty. Generic models were proposed for single and multiple allocation problems. Reference [49] also presented a programming formulation for hierarchical multimodal hub location problem with time-definite deliveries and analyzed the sensitivity on Turkish network. Reference [50] studied the hub location problem under competition with Civil Aeronautics Board (CAB) and Turkish Postal (TR) datasets as case studies.

In Tables 1 and 2, a summary of allocation problems and representative solution techniques in the literature is shown. Most of other papers in the references that are covered by them are not shown here. Various methods have been proposed for solving hub location problems, such as LR, GA, and CBS. Methods are often tuned for specific datasets, specific hardware properties, or specific parameters. Although newly published methods compare with few prior works, the selection of these works is often suboptimal, and comparisons are carried out on different datasets with different parameters; datasets are not often made publicly available, which means that results are not reproducible. Moreover, getting an overview over the state of the art in this field is difficult, as results are published in various domains without much cross-talk and often study variations of the core problems. It is difficult for new researchers to choose the best solution algorithm for a given problem instance, because of the lack of common benchmarks and the dispersal of research in different communities caused by the heterogeneity of problems and approaches.

In this paper, the formulations of five types of p-hub median problem variations (classical single, classical multiple, reliable single, reliable multiple, and incomplete single allocation problems) are surveyed first. Then, three different solution techniques from the state of the art, GA, LR, and restricted CBS (RCBS) method, for solving p-hub median problems are provided. In addition, we use CPLEX as a benchmark reference. Finally, to compare the performance of these methods, the commonly used TR dataset [51], CAB dataset [12], and Australia Post (AP) dataset [52] are selected as case studies. In our experiment, the solution qualities, computation time, and main memory usage are evaluated.

We would like to emphasize that this paper is not intended as a theory-focused survey of solution techniques. There exist plenty of excellent works in this area [9–11]. The major contributions of our study are as follows:

Table 2: Comparison of solution techniques in the literature: a method is marked with "√" in the corresponding row, if it was compared in a paper. The methods marked with "O" are proposed by the researchers in the first column. See Abbreviation Summary of Methods for abbreviations of solution techniques.

	TTS	SAA	BB	LR	BP	GA	CPLEX	PRM	WA	LAA	EBD	AP	RCA	CBS
[21]	O													
[22]		O	O											
[23]		√	√	O										
[27]		√				O								
[28]	√	√				O		√						
[18]							O							
[29]											O	√	√	
[24]				O	O		√							
[30]						O	√							
[26]			O	O			√							
[31]						O			√	√				
[32]														O
[33]						O	√							
[34]											O			

(1) We perform an exhaustive evaluation of five types of hub location problems and four solution techniques. We synthesize the essence of solution algorithms and adapt them for different problem instances, to obtain five competitors for each method.

(2) All competitors are evaluated against three standard datasets from the literature: TR, CAB, and AP. Running all experiments with the same setup (hardware/software) parameters allows us to provide a comprehensive evaluation of all techniques. Moreover, we provide an unbiased view on the state of the art, as it is perceived by the papers and their methodological descriptions.

(3) The wealth of experiments we performed and the number of techniques we compared allow us to draw several insights into scalability, solution quality, and the possibility of exploiting parallelism.

We would like to point out that a comparison of different methodologies has to be taken with caution. Some methodologies were proposed to find provably optimal solutions, while others are pure heuristics without any guarantees on optimality. Comparing their solution quality and solution time allows us to estimate the trade-off between running time and solution quality by each method for different datasets. This view is novel to the literature, where, usually, methodologies are not cross-compared. In addition, we have implemented several standard solution techniques in the literature, a considerable amount of work, given that the source code is not published alongside of papers. Clearly, we implemented only a subset of existing methods. Finally, the hub location models chosen for our study make simplified assumptions about the network and the modeling level of detail, for instance, ignoring delay and propagation of delay through a network [53]. Therefore, our study should be understood as a first step into the comparative evaluation of hub location problems.

The remainder of this paper is organized as follows. The formulations of five types of hub location problem models are provided in Section 2. Selected solution techniques, that is, genetic algorithm (GA), Lagrangian relaxation (LR), and clustering-based (CBS) method, are abstracted and synthesized in Section 3. To compare these methods, the evaluation of TR dataset, AP dataset, and CAB dataset as case studies is presented in Section 4. The paper concludes with Section 5.

2. p-Hub Median Problem

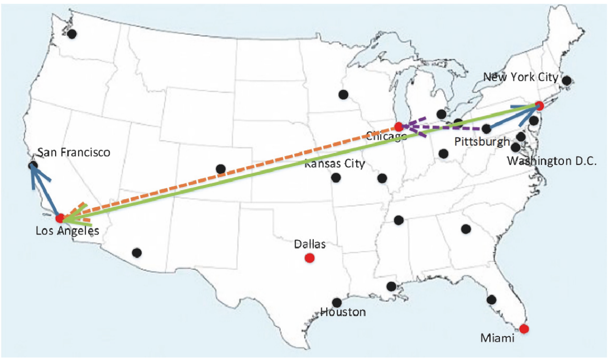


Figure 2: Hub-and-spoke structure and alternative routes: there exists a regular route from Pittsburgh to San Francisco with New York City and Los Angeles as hubs. If hub of New York City is disrupted, an alternative route "Pittsburgh → Chicago → Los Angeles → San Francisco" can be used. Note that, the Los Angeles hub is still in the alternative route.

The p-hub median problems discussed in this study have two distinct structures: single allocation (SA) and multiple allocation (MA). In addition, each problem can also be considered in the classical models and as a reliable model, depending on whether the reliability of hubs is taken into account. In classical models, all hubs are assumed to function properly, while each hub can be disrupted in the reliable models [54]. An example is shown in Figure 2: There is a regular route from Pittsburgh to San Francisco with two hubs, New York City and Los Angeles. If New York City hub is disrupted, an alternative route "Pittsburgh → Chicago → Los Angeles → San Francisco" can be used, and node Chicago is the backup hub of New York City. At last, the single allocation problem with incomplete hub networks is also considered. All five formulations of p-hub median problems are introduced and formalized in this section: classical single allocation (CSA), classical multiple allocation (CMA), reliable single allocation (RSA), reliable multiple allocation (RMA), and incomplete single allocation (ISA). The major reason for revisiting the mathematical formulation is that many slightly different problem formulations have been used across existing studies, where modifications often have a significant impact on expressivity and time complexity.

2.1. Classical Single Allocation Problem. In classical single allocation problem, each node is allocated to a single hub and all the inbound or outbound flows of each node must go through its hub. Let $G = (V, E)$ be a network. Here V and E are the set of nodes and links, respectively. The numbers of nodes and hubs are n and p. Let c_{ij} and w_{ij} be the cost and the flow between nodes i and j for each pair of (i, j). The formulation for CSA is shown as follows [55].

CSA-P

$$\min \quad \sum_{i\in V}\sum_{k\in V}\sum_{m\in V}\sum_{j\in V, j<i} w_{ij}F_{ikmj}X_{ikmj} \tag{1}$$

$$\text{subject to} \quad \sum_{m\in V} X_{ikmj} = Y_{ik}, \quad \forall i,\ j<i,\ k\in V \tag{2}$$

$$\sum_{k\in V} X_{ikmj} = Y_{jm}, \quad \forall i,\ j<i,\ m\in V \tag{3}$$

$$\sum_{k\in V} Y_{ik} = 1, \quad \forall i\in V \tag{4}$$

$$\sum_{k\in V} Y_{kk} = p \tag{5}$$

$$Y_{ik} \le Y_{kk}, \quad \forall i,k\in V \tag{6}$$

$$Y_{ik} \in \{0,1\}, \quad \forall i, k \in V \tag{7}$$

$$X_{ikmj} \in \{0,1\}, \quad \forall i, k, m, \ j < i \in V. \tag{8}$$

Here, (1) is the objective function of total costs. Let $F_{ikmj} = c_{ik} + \alpha c_{km} + c_{mj}$ be the cost of route (i, k, m, j) [56]. Parameter $\alpha < 1$ is the cost coefficient (economies of scale discount factor, see, e.g., [57]) for transporting the flows between two hubs. Variables X_{ikmj} and Y_{ik} are the route variable and assignment variable, respectively. They are defined as follows [58]:

$$Y_{ik} = \begin{cases} 1, & \text{if } i \text{ is assigned to hub } k \\ 0, & \text{otherwise,} \end{cases}$$

$$X_{ikmj} = \begin{cases} 1, & \text{if flow on } (i, j) \text{ is routed through } k \text{ and } m \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

Equations (2)-(3) ensure that if node i is assigned to hub k (or j assigned to m), the flow from i (or to j) must go through hub k (or m). Equation (4) ensures that each node can be associated with one hub only. Equation (5) indicates the number of hubs is p. Equation (6) indicates that each node can only be assigned to a hub. Based on the fundamental formulation for CSA presented by [59], [22] also formulated CSA with a new model, in which they reduced the number of variables for linear program from $O(n^4)$ to $O(n^3)$. Finally, an assumption has been used in many studies: $w_{ij} = w_{ji}, \forall i, j \in V$. Therefore, in this paper, we consider only the case of $j < i$.

2.2. Classical Multiple Allocation Problem. In classical multiple allocation problem, the flows from one origin to different destinations can be routed through different hubs. Thus, we do not need to assign each node to a specific hub. Let Y_k be the variable that is used to indicate whether node k is a hub, and the route variable X_{ikmj} is still used. The formulation for the CMA problem is shown as follows [45, 60].

CMA-P

$$\min \sum_{i \in V} \sum_{k \in V} \sum_{m \in V} \sum_{j \in V, j<i} w_{ij} F_{ikmj} X_{ikmj} \tag{10}$$

$$\text{subject to} \quad \sum_{m \in V} X_{ikmj} \leq Y_k, \quad \forall i, \ j < i, \ k \in V \tag{11}$$

$$\sum_{k \in V} X_{ikmj} \leq Y_m, \quad \forall i, \ j < i, \ m \in V \tag{12}$$

$$\sum_{k \in V} \sum_{m \in V} X_{ikmj} = 1, \quad \forall i, \ j < i \in V \tag{13}$$

$$\sum_{k \in V} Y_k = p \tag{14}$$

$$Y_k \in \{0,1\}, \quad \forall k \in V \tag{15}$$

$$X_{ikmj} \in \{0,1\}, \quad \forall i, k, m, \ j < i \in V. \tag{16}$$

Here, (11)-(12) ensure that the route for each OD pair (i, j) must go through hubs. Equation (13) shows that each OD pair (i, j) can have only one route.

2.3. Reliable Single Allocation Problem. The problems above are both without the consideration of hub reliability. In real-world cases, however, each hub can be disrupted for many reasons. For instance, in air transportation networks, nodes (i.e., airports) can be disrupted due to bad weather or equipment outage. In this case, backup hubs need to be considered. Let $q_k \in [0, 1]$ be the disruption probability of node k. Let q_m^k be q_m if $m \neq k$ and 0 otherwise. For the backup assignment, we use the strategy proposed by [26]. They made two major assumptions: first, only single disruptions are considered; second, for the routes with two hubs, the unaffected hub must still be inside the alternative route. Assume U_{iju} and V_{iju} are the backup hub variables that are defined as follows:

$$U_{iju} = \begin{cases} 1, & \text{if hub } u \text{ is the backup hub for the first hub of OD pair } (i, j) \\ 0, & \text{otherwise} \end{cases}$$
$$V_{iju} = \begin{cases} 1, & \text{if hub } u \text{ is the backup hub for the second hub of OD pair } (i, j) \\ 0, & \text{otherwise.} \end{cases} \tag{17}$$

Therefore, the RSA problem is formulated as follows [26].

RSA-P

$$\min \sum_{i \in V} \sum_{k \in V/\{i\}} \sum_{m \in V} \sum_{j \in V/\{m\}, j<i} w_{ij} F_{ikmj} \left(1 - q_k - q_m^k\right) X_{ikmj}$$
$$+ \sum_{i \in V} \sum_{j \in V, j<i} w_{ij} \left(\sum_{m \in V/\{j\}} F_{iimj} \left(1 - q_m^i\right) X_{iimj} + \sum_{k \in V/\{i\}} F_{ikjj} \left(1 - q_k^j\right) X_{ikjj} + F_{iijj} X_{iijj} \right)$$

$$
\begin{aligned}
&+ \rho \sum_{i \in V} \sum_{j \in V, j<i} w_{ij} \left(\sum_{k \in V/\{i,j\}} \sum_{m \in V/\{i\}} \sum_{u \in V} F_{iumj} q_k X_{ikmj} U_{iju} \right. \\
&\left. + \sum_{k \in V/\{j\}} \sum_{m \in V/\{i,j\}} \sum_{u \in V} F_{ikuj} q_m X_{ikmj} V_{iju} + \sum_{k \in V/\{i,j\}} \sum_{u \in V} F_{iuuj} q_k X_{ikkj} U_{iju} \right)
\end{aligned} \tag{18}
$$

$$
\text{subject to} \quad \sum_{m \in V} X_{ikmj} = Y_{ik}, \quad \forall i,\ j < i,\ k \in V \tag{19}
$$

$$
\sum_{k \in V} X_{ikmj} = Y_{jm}, \quad \forall i,\ j < i,\ m \in V \tag{20}
$$

$$
\sum_{k \in V} Y_{ik} = 1, \quad \forall i \in V \tag{21}
$$

$$
\sum_{k \in V} Y_{kk} = p \tag{22}
$$

$$
U_{ijk} + Y_{ik} \leq Y_{kk}, \quad \forall i,\ j < i,\ k \in V \tag{23}
$$

$$
V_{ijm} + Y_{jm} \leq Y_{mm}, \quad \forall i,\ j < i,\ m \in V \tag{24}
$$

$$
\sum_{k \in V} U_{ijk} = 1 - Y_{ii} - Y_{ij}, \quad \forall i,\ j < i \in V \tag{25}
$$

$$
\sum_{m \in V} V_{ijm} = 1 - Y_{jj} - Y_{ji}, \quad \forall i,\ j < i \in V. \tag{26}
$$

Here, the first two terms of the objective function are the cost for regular routes and the left term is the cost for alternative routes. Equations (19)–(22) are the same as (2)–(5). Equations (23)-(24) ensure that nodes can be assigned only to hub nodes, and backup nodes must be different from the corresponding hubs. Equations (25)-(26) indicate that one hub needs an alternative hub only if it is different from the OD pair (i, j) and it cannot be disrupted otherwise. It shows that *RSA-P* is a quadratic integer program. It can be linearized with the linearization method proposed by [61].

2.4. Reliable Multiple Allocation Problem. Similar to Section 2.2, variable Y_k is used to denote whether node k is a hub. Therefore, the formulation of the RMA problem is shown as follows [26].

RMA-P

$$
\begin{aligned}
\min \quad & \sum_{i \in V} \sum_{k \in V/\{i\}} \sum_{m \in V} \sum_{j \in V/\{m\}, j<i} w_{ij} F_{ikmj} \left(1 - q_k - q_m^k\right) X_{ikmj} \\
&+ \sum_{i \in V} \sum_{j \in V, j<i} w_{ij} \left(\sum_{m \in V/\{j\}} F_{iimj} \left(1 - q_m^i\right) X_{iimj} + \sum_{k \in V/\{i\}} F_{ikjj} \left(1 - q_k^j\right) X_{ikjj} + F_{iijj} X_{iijj} \right) \\
&+ \rho \sum_{i \in V} \sum_{j \in V, j<i} w_{ij} \left(\sum_{k \in V/\{i,j\}} \sum_{m \in V/\{i\}} \sum_{u \in V} F_{iumj} q_k X_{ikmj} U_{iju} \right. \\
&\left. + \sum_{k \in V/\{j\}} \sum_{m \in V/\{i,j\}} \sum_{u \in V} F_{ikuj} q_m X_{ikmj} V_{iju} + \sum_{k \in V/\{i,j\}} \sum_{u \in V} F_{iuuj} q_k X_{ikkj} U_{iju} \right)
\end{aligned} \tag{27}
$$

$$
\text{subject to} \quad \sum_{m \in V} X_{iimj} = Y_i, \quad \forall i,\ j < i \in V \tag{28}
$$

$$\sum_{k\in V} X_{ikjj} = Y_j, \quad \forall i,\ j < i \in V \tag{29}$$

$$\sum_{k\in V}\sum_{m\in V} X_{ikmj} = 1, \quad \forall i,\ j < i \in V \tag{30}$$

$$\sum_{k\in V} Y_k = p \tag{31}$$

$$U_{ijk} + \sum_{m\in V} X_{ikmj} \le Y_k, \quad \forall i,\ j < i,\ k \in V \tag{32}$$

$$V_{ijm} + \sum_{k\in V} X_{ikmj} \le Y_m, \quad \forall i, j, m \in V \tag{33}$$

$$\sum_{k\in V} U_{ijk} = 1 - Y_i - \sum_{m\in V} X_{ijmj}, \quad \forall i,\ j < i \in V \tag{34}$$

$$\sum_{m\in V} V_{ijm} = 1 - Y_j - \sum_{k\in V} X_{ikij}, \quad \forall i,\ j < i \in V. \tag{35}$$

Here, (28)-(29) ensure that each hub node can only be assigned to itself. Equations (32)–(35) are similar to (23)–(26).

2.5. Incomplete Single Allocation Problem. For the incomplete single allocation problem, in addition to hub location and assignment, a fixed number q hub links needs to be determined. The formulation for this problem is quite different from the formulations introduced above. Let $O_i = \sum_j w_{ij}$ be the total flow from source node i, $D_j = \sum_i w_{ij}$ be the total flow to destination node j, and f^i_{km} be the flow originating from node i which is routed on hub link (k, m). The decision variable Z_{ij} for hub links is defined as follows:

$$Z_{ij} = \begin{cases} 1, & \text{if hub } i \text{ and hub } j \text{ are connected} \\ 0, & \text{otherwise.} \end{cases} \tag{36}$$

Therefore, the ISA problem is formulated as follows [18].

ISA-P

$$\min \quad \sum_{i\in V}\sum_{k\in V} c_{ik} O_i Y_{ik} + \sum_{i\in V}\sum_{k\in V} c_{ki} D_i Y_{ik} + \alpha \sum_{i\in V}\sum_{k\in V}\sum_{m\in V} c_{km} f^i_{km} \tag{37}$$

$$\text{subject to} \quad \sum_{k\in V} Y_{ik} = 1, \quad \forall i \in V \tag{38}$$

$$\sum_{k\in V} Y_{kk} = p \tag{39}$$

$$\sum_{k\in V}\sum_{m\in V, m<k} Z_{km} = q \tag{40}$$

$$Y_{ik} \le Y_{kk}, \quad \forall i, k \in V \tag{41}$$

$$Z_{km} \le Y_{kk}, \quad \forall k,\ m < k \in V \tag{42}$$

$$Z_{km} \le Y_{mm}, \quad \forall k,\ m < k \in V \tag{43}$$

$$\sum_{m\in V, m\neq k} f^i_{mk} + O_i Y_{ik} = \sum_{m\in V, m\neq k} f^i_{km} + \sum_{j\in V, j<i} w_{ij} Y_{jk}, \quad \forall i, k \in V \tag{44}$$

$$f^i_{km} + f^i_{mk} \le O_i Z_{km}, \quad \forall i,\ j < i,\ k \in V \tag{45}$$

$$f^i_{km} \ge 0, \quad \forall i, k,\ m \neq k \in V \tag{46}$$

$$Y_{ik} \in \{0, 1\}, \quad \forall i, k \in V \tag{47}$$

$$Z_{km} \in \{0, 1\}, \quad \forall k,\ m < k \in V. \tag{48}$$

In (37), the first two terms are the cost of flows between hubs and nonhub nodes, and the third term is the cost of flows in hub networks. Equations (38)-(39) are the constraints of single allocation and number of hubs. Equation (40) is the constraint for the number of hub links; that is, q links are established between hubs. Equations (41)–(43) indicate that nodes can be assigned to a hub only and hub links can be established between hubs only. Equations (44)-(45) are the flow equilibrium constraint and flow capacity constraint, respectively.

3. Solution Techniques

In Section 2, we revisited the formulations of five different p-hub median problems. To solve these problems, various solution techniques have been proposed, such as GA and LR; see the introduction and relevant surveys for details. In most literature, however, researchers usually compared their approaches with very few/outdated methods instead of new algorithms from the state of the art. Furthermore, the properties of some new methods were compared only with certain optimization solvers (e.g., CPLEX). Several approaches from the state of the art are summarized and revised accordingly in this section, such as GA (Section 3.2), LR (Section 3.3), and CBS (Section 3.4), all tailored towards the five standard problem definitions.

3.1. Overview of Solution Techniques. In this section, the idea of each method for solving five p-hub median problems is revisited. Let us start with GA. A large number of near-optimal solutions can be obtained with *GA*, which has been used to solve many complicated problems and its feasibility for solving hub locations problems has been shown in the literature. When solving hub location problems with GA, chromosomes are used to represent solutions [62]. In the CSA problem, each node is allocated to one hub. Thus, node assignments can be modeled as chromosomes directly. The crossover and mutation operators are used to change the node assignments and generate new solutions. For the CMA problem, the chromosomes are represented by the routes between OD pairs. In addition, in the reliable cases, we need to add an operator for selecting the backup hubs, which is used to find an optimal alternative route for each given regular route. For incomplete problem, the connectivity of hub networks is considered first while establishing hub links. In *LR*, the formulations of five allocation problems can be transformed into two independent subproblems, by adding several Lagrangian multipliers. Then, the lower bounds and upper bounds of the original problem can be obtained by solving these subproblems. The Lagrangian multipliers can be updated iteratively, until reaching the optimal solution. The *CBS method* is a very novel approach for analyzing hub location problem. This method was proposed for solving single allocation problems. However, it can also be applied to multiple cases. In this method, based on spatial properties and travel demands of nodes, a subset of potential hubs is obtained. Then, several constraints based on these subsets are added to the problem formulations, and the computational complexity is reduced significantly.

Note most solution methods that have been proposed by researchers. We use the original procedures to describe them. However, if a solution method has not been published for solving a given problem, we do some modification based on previous versions of the method. For instance, GA has not yet been published for solving incomplete problems and the procedure of this method is modified for incomplete problems in our study.

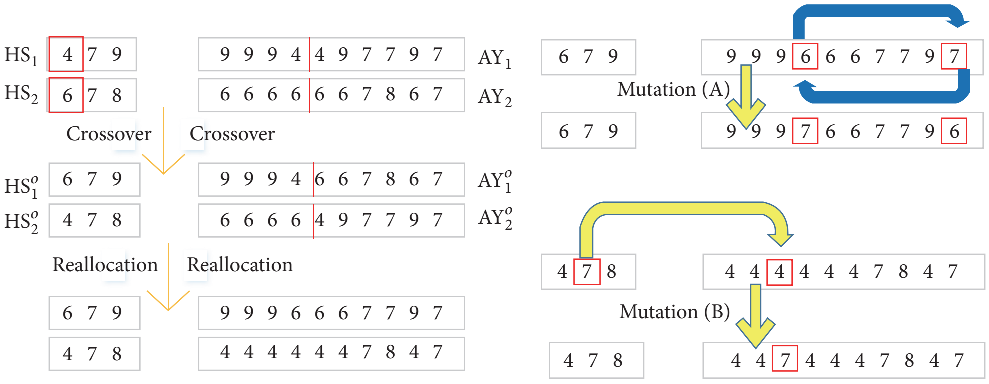
3.2. Genetic Algorithms. To solve hub location problems, several researchers presented GA-based methods in recent years. Because most methods are proposed for solving single allocation problems, the standard strategy for CSA using GA is introduced first. Then, based on this simple strategy, we develop an extension for RSA by adding a method for the backup hub selection. The GA strategy developed in this section is based mostly on the approaches proposed by [27, 33].

Initial Population. In the single allocation problem, each node is assigned to a particular hub. Thus, each individual has two arrays: hub array (HS) and assignment array (AY). The hub array HS has a length of p and represents the hub set, and the assignment array AY has a length of n and represents the assignment of nodes to hubs. If node i is assigned to hub k, then $\mathrm{AY}[i] = k$. Here we assume that if node k is a hub, then it must be assigned to itself; that is, $\mathrm{AY}[k] = k$. Based on the solution representation above, the initial population can be generated. However, before that, the hub set must be determined. To reduce the total cost, large flows should travel through the routes with less cost. Let $O_i = \sum_{j\in V, j<i} w_{ij}$ and $D_i = \sum_{j\in V, j<i} w_{ji}$ be the total outbound flow and inbound flow of node i, respectively. Thus, it should be more likely to locate hubs at those nodes with large flows to travel (i.e., i with large $O_i + D_i$). Then, in each chromosome, we assign each node to its closest hub.

Crossover. With initial population set, new offspring chromosomes can be generated by the crossover operator. To choose the parent chromosomes, we use roulette wheel sampling [27] and the fitness value of each individual is represented by the reciprocal of the total cost of the solution. The crossover operator is introduced with a small example. As shown in Figure 3, two parent chromosomes are selected randomly and the number of nodes and hubs is $n = 10$ and $p = 3$, respectively. First, we choose a swapping number n_h from 1 to $p/2$ and a crossover point n_a from 1 to n, randomly. Then, n_h hubs are swapped between the two hub arrays. Each assignment array can be divided into two parts by the point n_a, and each part is combined with the opposite part of the other assignment array. For instance, in this example, we assume that $n_h = 1$ and $n_a = 4$. One hub is swapped between HS_1 and HS_2. If they are 4 and 6, then new hub arrays are $\mathrm{HS}_1^o = [6, 7, 9]$ and $\mathrm{HS}_2^o = [4, 7, 8]$. Next, we combine the left part of AY_1 with the right part of AY_2, and new assignment array AY_1^o is generated. With the same operator, AY_2^o can also be obtained. Now, there may exist an infeasible case where some nodes are assigned to a nonhub node. For instance, in AY_1^o, node 4 is assigned to itself which is not a hub in the new individual. Therefore, to avoid the infeasibility, we need to do a reallocation after each crossover operator. Each node assigned to a nonhub node needs to be reallocated to its closest hub in the new hub array. In addition, each hub must be reallocated to itself, obviously.

Mutation. To avoid the early convergence near a local optimal solution, the mutation needs to be operated on the chromosomes with a probability. In this paper, two mutation operators are applied to the assignment arrays only: (A) Select two nonhub nodes randomly and swap their assignment. (B) Select one nonhub node randomly and reallocate it to another hub.

Because each hub node must be assigned to itself, the mutation operator is applied only to nonhub nodes (i.e., spokes). For each chosen individual, these two mutation operators are both used, and the result with more total cost is discarded. Using a crossover operator and a mutation operator, two offspring individuals can be generated. Then, elitism is applied in the population. The individuals with worse fitness values are replaced by those with better fitness values.



(a) Crossover

(b) Mutation

Figure 3: A small example of crossover and mutation. Hubs are encoded by numbers: (a) Hubs 4 and 6 are swapped between two hub arrays. The assignments of the nodes after the red lines are also swapped between two assignment arrays. After the reallocation, the offspring individuals are generated. (b) The assignments of two spokes are exchanged in the first mutation; a random spoke is reassigned to a random hub in the second mutation.

Backup Hub Selection. With the GA strategy introduced above, the CSA problem can be solved. However, while solving the RSA problem, the backup hubs need to be considered after determining the hub set and assignment. Because the alternative route of each OD pair (i, j) is independent, we can find the best backup hubs for each individual (i, j) by considering the route cost. For each OD pair (i, j), the regular route (i, k, m, j) is determined by GA. Then, the best backup hubs for k can be found by computing the costs of all possible backup hubs which is different from k and selecting the minimum one. A similar strategy is used for selecting the best backup hub for m.

GA Strategy for Multiple Allocation Problems. Each node is assigned to one hub in single allocation problems. However, in multiple allocation problems, each node can be assigned to different hubs for different OD pairs. Thus, the chromosome size is equal to the number of OD pairs here. Because we only consider the case $j < i$, the size is equal to $n(n-1)/2$. While generating the initial population, we use the same method as single allocation problem to construct the hub array. Then for each (i, j), where $i, j \in V$ and $j < i$, we choose a pair of hubs (k, m) as its regular hubs from the hub array randomly and the backup hubs can also be selected with the method above similarly.

In the crossover step, we sort OD pairs in the order of $[(1,2),(1,3),\ldots,(1,n),(2,3),(2,4),\ldots,(n-2,n-1),(n-2,n),(n-1,n)]$ and OD pair (i, j) has an order number $((2n-i)(i-1)/2+j-i)$. Then, we choose two parent chromosomes with roulette wheel sampling and the hub arrays are swapped with the crossover method introduced above. After selecting a crossover point n_a randomly, all the OD pairs with an order number less than n_a are swapped between two parent chromosomes. Then, some OD pairs may use nonhub nodes as their hubs. Therefore, these OD pairs need to be reallocated to their best hubs $[k^*, m^*]$ with the least cost. After the reassignment, the last offspring individuals are obtained. In the mutation operator, a nonhub node i and a hub k are chosen randomly. Then all first hubs (or second hubs) of OD pairs $(i, j)(j < i)$ (or $(j, i)(j > i)$) are set to hub k.

GA Strategy for ISA Problem. GA has not yet been published for solving ISA problems and we propose a new strategy here. In addition to the hub array and assignment array, the hub links should also be determined in the ISA problem. Thus, a set of hub links is added to each individual. We use the same method as single allocation problem to construct the hub array and assignment array while generating the initial population. In order to guarantee the connectivity of the hub network, a randomized version of the Prim algorithm [63] is used to generate hub links (See Algorithm 1).

In crossover step, after obtaining offspring hub sets with the same strategy as Figure 3, the link sets of offspring can be generated as follows: Let HUB_1^o and HUB_2^o be hub sets of two offspring. Let Link_1 and Link_2 be the link sets of two parent chromosomes. Then, $\text{Link}_d = \text{Link}_1 \cap \text{Link}_2$, $\text{Link}_u = \text{Link}_1 \cup \text{Link}_2$, and $\text{Link}_{c1} = \text{Link}_u \cap \{\text{links in complete graph in HUB}_1\}$. For the first offspring, we remove several links from Link_{c1} with the guarantee of connectivity of HUB_1 network. The remaining q links construct the link set of the first offspring individual. If Link_{c1} is not connective or the number of its links is less than q, we can add several links to it randomly first. The same strategy is applied to the second offspring. In the mutation step, in addition to the operation on assignment array, a link is selected from the link set randomly and replaced by another random link on the basis of connectivity of hub network.

3.3. Lagrangian Relaxation. As a widely used algorithm, Lagrangian relaxation has also been applied to hub location problems [23, 26]. In this section, we introduce the procedure of LR for solving hub location problems mostly based on the formulation *CSA-P* (see (1)–(8)) presented in Section 2.1. The methods for solving the other three problems are similar.

```
Input hub set $H$, empty link set $L_1 = \emptyset$, complete link set $L_2 = \{$links in complete graph of $H\}$,
empty hub set $V_1 = \emptyset$, set $V_2 = H$, number of hubs $p$, number of hub links $q$.
if $q \geq \frac{p(p-1)}{2}$:
    $L_1 = \{$links in complete graph of $H\}$
else:
    choose a hub $u$ from $V_2$ randomly
    $V_1 = V_1 \cup \{u\}$
    $V_2 = V_2 \backslash \{u\}$
    while $V_1 \neq H$:
        choose a hub $u$ from $V_1$ and a hub $v$ from $V_2$ randomly
        generate a link $(u, v)$
        $L_1 = L_1 \cup \{(u, v)\}$
        $L_2 = L_2 \backslash \{(u, v)\}$
        $V_1 = V_1 \cup \{v\}$
        $V_2 = V_2 \backslash \{v\}$
    endwhile
    choose $q - p + 1$ links from $L_2$ randomly, marked as $L_3$
    $L_1 = L_1 \cup L_3$
endif
Output the set hub hub links $L_1$
```

Algorithm 1: Random Prim algorithm for finding a connective hub network.

Lower Bound. Let η_{ijk}, β_{ijm}, γ_i, $(i,\ j < i,\ k \in V)$ be the Lagrangian multipliers of (2)–(4). Then *CSA-P* can be transformed to a Lagrangian problem as follows:

$$L(\eta, \beta, \gamma) = \min \sum_{i \in V} \sum_{k \in V} \overline{C_{ik}} Y_{ik} - \sum_i \gamma_i + \sum_{i \in V} \sum_{j \in V, j<i} \sum_{k \in V} \sum_{m \in V} \overline{F_{ikmj}} X_{ikmj} \quad (49)$$

$$\text{subject to} \quad (5)\text{–}(8), \quad (50)$$

where

$$\overline{C_{ik}} = \gamma_i - \sum_{j \in V, j<i} \eta_{ijk} - \sum_{j \in V, j<i} \beta_{ijk},$$
$$\overline{F_{ikmj}} = w_{ij} F_{ikmj} + \eta_{ijk} + \beta_{ijm}. \quad (51)$$

Note that two groups of variables Y_{ik} and X_{ikmj} are independent now. Thus, the Lagrangian problem can be decomposed into two subproblems as follows.

LSub1

$$\min \sum_{i \in V} \sum_{k \in V} \overline{C_{ik}} Y_{ik} - \sum_i \gamma_i. \quad (52)$$
$$\text{subject to} \quad (5)\text{–}(7)$$

LSub2

$$\min \sum_{i \in V} \sum_{j \in V, j<i} \sum_{k \in V} \sum_{m \in V} \overline{F_{ikmj}} X_{ikmj} \quad (53)$$

$$\text{subject to} \quad \sum_{k \in V} \sum_{m \in V} X_{ikmj} = 1, \quad \forall i,\ j < i \in V. \quad (54)$$

```
for node $i$ in $V$:
    if $\sum_{j \in V} \leq 1$:
        find the nearest hub $k$ from node $i$
        set $Y_{ij} = 0, \forall j \in V$
        set $Y_{ik} = 1$
    endif
endfor
```

Algorithm 2: Finding the upper bound for Lagrangian relaxation.

Here (54) is used to reduce the size of the solution set in the model, and the result qualities can be improved significantly. The lower bound of the *CSA-P* can be obtained by solving the two subproblems with the methods proposed by [23]. To obtain the optimal solution efficiently, we need a method for finding the upper bound.

Upper Bound. The upper bound can be obtained based on the solution of *LSub1*. After solving this subproblem, a set of p hubs and some assignments Y_{ik} are determined. However, in these results, some nodes may be assigned to more than one hub, and some nodes may be unassigned. Therefore, Algorithm 2 is used to find an upper bound with a feasible solution.

With the methods above, the lower bound and upper bound can be obtained for a group of particular Lagrangian multipliers. In this section, the complete procedure of LR with multiplier updating is shown as follows.

(a) Initialization includes the following: iteration number $r = 0$, max number of iterations N_{ite}, max gap M_{gap}, max iteration number for no improvement of lower bound N_{imp}, step size multiplier $\lambda = 6$, and initial Lagrangian multipliers $\eta, \beta, \gamma = 0$.

(b) Finding the lower bound LB is by solving *LSub1* and *LSub2*.

Finding the upper bound UB is with Algorithm 2.

(c) Updating Lagrangian multipliers is as follows [64]:

$$\begin{aligned} \eta_{ijk}^{r+1} &= \eta_{ijk}^{r} + \theta^r \left(\sum_{m\in V} X_{ikmj} - Y_{ik} \right), \\ &\qquad \forall i,\ j < i,\ k \in V, \\ \beta_{ijm}^{r+1} &= \beta_{ijm}^{r} + \theta^r \left(\sum_{k\in V} X_{ikmj} - Y_{jm} \right), \\ &\qquad \forall i,\ j < i,\ m \in V, \\ \gamma_{i}^{r+1} &= \gamma_{i}^{r} + \theta^r \left(\sum_{k\in V} Y_{ik} - 1 \right), \quad \forall i \in V, \end{aligned} \tag{55}$$

where

$$\begin{aligned} \theta^r &= \lambda \frac{\text{UB} - \text{LB}}{Q}, \\ Q &= \sum_{i,j,k} \left(\sum_{m} X_{ikmj} - Y_{ik} \right)^2 + \sum_{i,j,m} \left(\sum_{k} X_{ikmj} - Y_{jm} \right)^2 \\ &\quad + \sum_{i} \left(\sum_{k} Y_{ik} - 1 \right)^2 . \end{aligned} \tag{56}$$

(d) If the lower bound LB is not improved in N_{imp} continuous iterations, then $\lambda = \lambda/2$, and η, β, γ also need to be set to the current best values.

(e) If iteration number $r = N_{\text{ite}}$ or $(\text{UB} - \text{LB})/\text{UB} < M_{\text{gap}}$, stop; else, go to (b).

The CSA problem can be solved with the procedure above. CMA, RSA, and RMA are solved with similar strategies proposed by [23, 26]. LR-based technique has not yet been published for solving ISA problems and, in fact, it is nontrivial to develop such a method. We need to define Lagrangian multipliers for several constraints and construct the Lagrangian problems. The constraint like (54) that can increase the convergence speed is necessary to be added to the Lagrangian problems but difficult to construct. Therefore, LR for the ISA problem is not evaluated in our study.

3.4. Clustering-Based Method. Reference [32] proposed a novel method for analyzing hub location problems by considering clustering-based potential hub sets. They define the travel demand of node i as $(O_i + D_i)$. Then, the following observations are obtained: (1) Nodes with larger travel demands are more likely to be chosen as hubs, especially those nodes far away from other nodes. However, some of the nodes are rarely chosen as hubs in the optimal solutions. (2) In a cluster of nodes that are close to each other, at most one of them can be selected as a hub usually, although the travel demand of this hub is not the largest. (3) The hubs prefer to be spread across the whole network. (4) The optimal hub locations are almost insensitive to the value of α.

Thus, the solutions of hub location problems strongly depend on both the spatial properties and the travel demands of nodes. Based on these properties, a heuristic is developed for generating a subset of potential hubs. The hubs for the optimal solutions can be selected from this subset and the computational complexity of the problem can be reduced. To generate the potential hub sets, all nodes need to be sorted by their importance, since the node importance depends on spatial properties and travel demands. The node importance can be measured with different formulations, and one of the best measures identified by [32] is as follows:

$$\text{Imp}_i = (O_i + D_i) * C_i, \tag{57}$$

where $C_i = \sum_{j\in V} c_{ij}$. Here, those nodes with large travel demands and far away from other nodes tend to be chosen as hubs. Then, we list all the nodes in descending order with the value of importance Imp_i and the potential hub set N^p is generated with $2p$ most important nodes. The hubs are more likely to be selected from N^p (note that if $2p \geq n$, then $N^p = V$).

Let $PM = \sum_{i\in N^p} (\min_{j\in N^p, j\neq i} c_{ij})/2p$ be the average shortest distances between nodes in N^p, and $N^p[t]$ is the tth most important node in N^p. Therefore, the clustering-based potential hub set algorithm (CBS) can be shown as Algorithm 3.

Here, each potential hub has a "hub circle" with a radius of PM. Node i will be assigned to set H_s, if there are no other nodes from N^p in its hub circle; node i will be assigned to set H, if there are two or more nodes from N^p that are less important than itself; node k will be assigned to set S_i, if k is within the hub circle of node i and node i is more important than k. Therefore, some subsets of potential hub nodes are generated based on the clustering properties, and several constraints for hub location can be added to the problem formulations as follows.

CBS

$$\begin{aligned} &\sum_{k\in S_i} Y_{kk} \geq 1, \quad \forall i \in H, \\ &\sum_{k\in H_s} Y_{kk} \geq 1. \end{aligned} \tag{58}$$

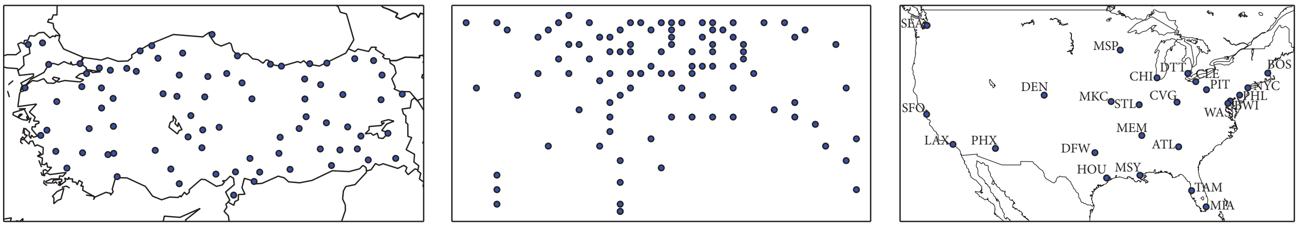
There must be at least one hub in each set S_i, $\forall i$, and at least one hub in set H_s. There is a restricted variant of CBS, called RCBS. In addition to (58), it adds another constraint where all the hubs must be selected from the potential hub sets, and the formulation is shown as follows:

$$\sum_{k\in\{\bigcup_{i\in H} S_i\}\cup H_s} Y_{kk} = p. \tag{59}$$

Note that CBS adds constraints independently of the problems at hand. Therefore, it is a prefiltering method for hubs and can be used for all problems without modification. In addition, there are several other variants of CBS, such as augmented CBS and improved RCBS. The details can be found in [32].

```
Input: $H = \emptyset$, $H_s = \emptyset$, $S_i = \emptyset, \forall i \in H$, $PM$, $t = 1$
while $t \le \min\{2p, n\}$:
    $flag = 0$
    for $i$ in $H$:
        if $N^p[t] \in S_i$:
            set $flag = 1$
            break
        endif
    endfor
    if $flag = 0$:
        $i = N^p[t]$
        set $S_i^o = \{k \in N^p : c_{ik} \le PM\}$
    endif
    if $\#S_i^o = 1$:
        $H_s = H_s \cup \{i\}$
    elif $\# S_i^o > 1$:
        $H = H \cup \{i\}$
        $S_i = S_i^o$
    endif
    $t = t + 1$
endwhile
```

Algorithm 3: Finding the clustering-based potential hub set (CBS).



(a) TR dataset with $n = 81$

(b) AP dataset with $n = 100$

(c) CAB dataset with $n = 25$

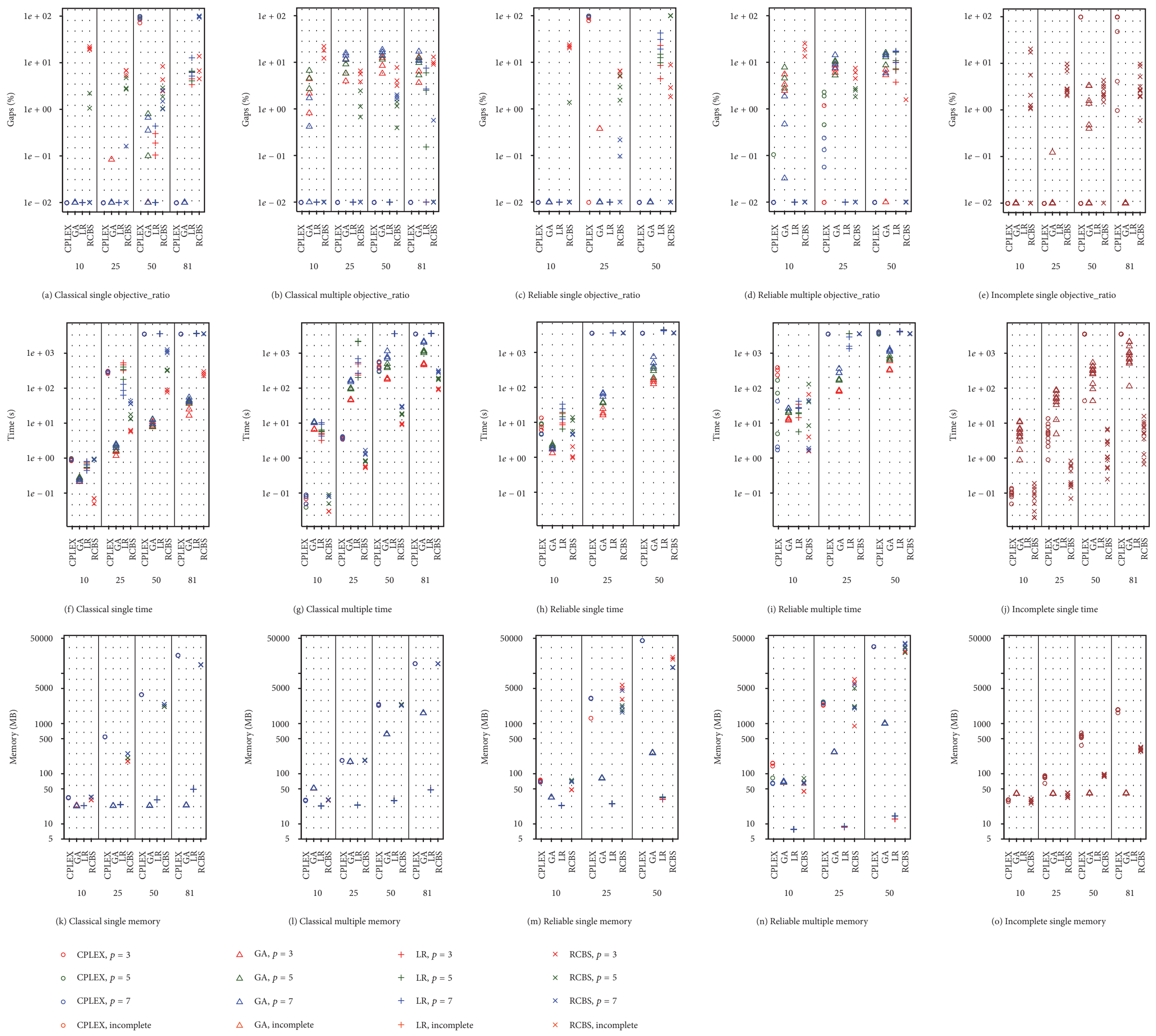
Figure 4: Visualization of three datasets used in this study: TR dataset, AP dataset, and CAB dataset.

4. Evaluation

In this section we report the results of our experimental evaluation of all implementations. Section 4.1 introduces the three datasets used in our study and describes the experimental setup. The results and analysis of the CSA, CMA, RSA, RMA, and ISA problems in the TR dataset are provided from Sections 4.2 to 4.6. In Section 4.7, we present the results for the AP dataset and CAB dataset that corroborate our findings for the TR dataset. The hub assignments for selected problem instances and datasets are visualized in Section 4.8. The analysis of parallel computation is shown in Section 4.9.

4.1. Introduction of Datasets and Experiments. To compare the performance of the methods introduced in Section 3, three commonly used datasets are utilized in this study. The TR (Turkish Postal) dataset includes 81 cities with distance and flows between them in Turkish Postal System [51]. The AP (Australia Post) dataset provides 200 nodes representing postcode districts with their coordinates and the flows between them in Australia [22]. The CAB (Civil Aeronautics Board) dataset is based the airline passenger interactions between 25 cities in USA in 1970 [12]. The numbers of nodes for the TR, AP, and CAB dataset are set to $n \in \{10, 25, 50, 81\}$, $n \in \{10, 25, 50, 100\}$, and $n \in \{10, 15, 20, 25\}$. We set the cost discount coefficient $\alpha \in \{0.3, 0.5, 0.7\}$ for the TR dataset and CAB dataset, as well as $\alpha = 0.75$ ($F_{ikmj} = 3c_{ik} + 0.75c_{km} + 2c_{mj}$) for the AP dataset. For complete (CSA, CMA, RSA, and RMA) problems, we set the numbers of hubs for three datasets $p \in \{3, 5, 7\}$. For ISA problems, we set $(p, q) \in \{(2, 1), (3, 2), (3, 3), (4, 4), (4, 5), (4, 6), (5, 6), (5, 7), (5, 8), (5, 9), (5, 10)\}$ [18]. The visualizations of all three datasets are shown in Figure 4. For the reliable models, the disruption probability q_k of each node k is set to a number randomly from $[0.01, 0.05]$. The rerouting coefficient ρ is set to 2.

We use the settings for GA and LR here as recommended in the literature. In GA, we set the population size $p_n = 200$ for single allocation problem and $p_n = 1000$ for multiple allocation problem. The maximum number of generations is 200 [27]. If the best fitness value cannot be improved within 10 consecutive generations, the algorithm will stop. In LR, the step size multiplier λ is set to 6, and the maximum iteration number is 3000 [23]. If the lower bound cannot be improved



(a) Classical single objective_ratio

(b) Classical multiple objective_ratio

(c) Reliable single objective_ratio

(d) Reliable multiple objective_ratio

(e) Incomplete single objective_ratio

(f) Classical single time

(g) Classical multiple time

(h) Reliable single time

(i) Reliable multiple time

(j) Incomplete single time

(k) Classical single memory

(l) Classical multiple memory

(m) Reliable single memory

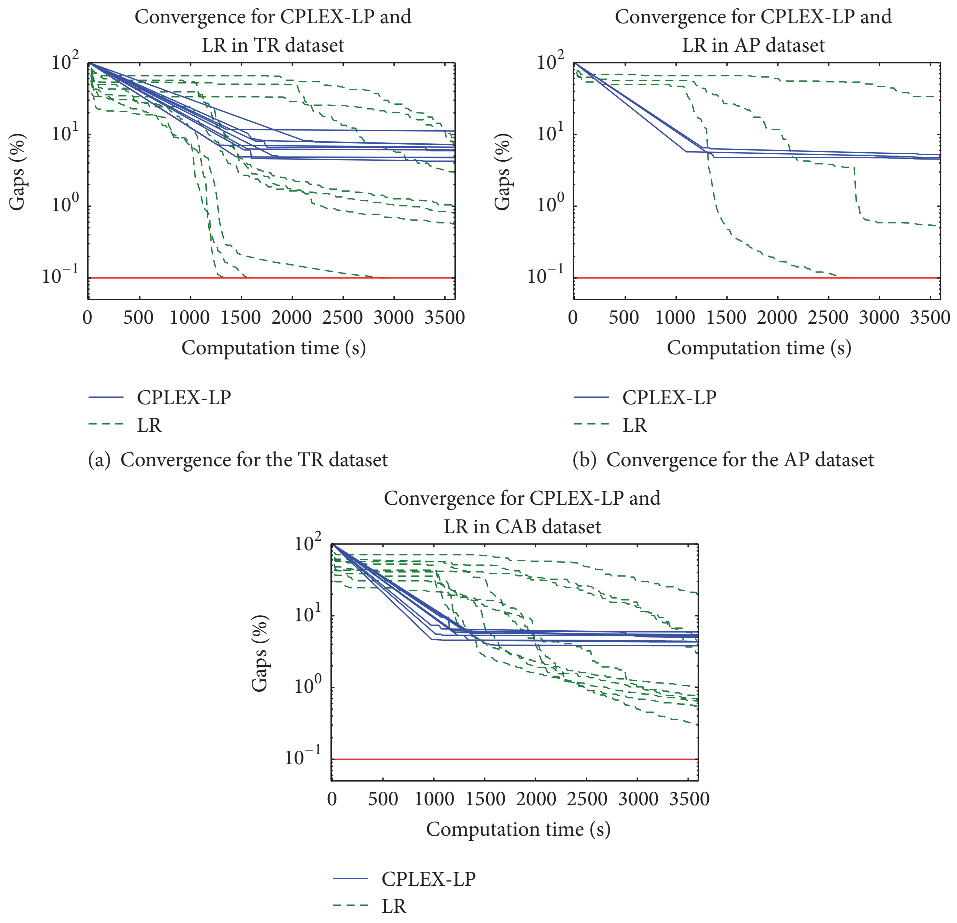
(n) Reliable multiple memory

(o) Incomplete single memory

Figure 5: Results of five types of problems for TR dataset. The problems in five columns are CSA, CMA, RSA, RMA, and ISA. The measurements of three rows are solution gaps, computation time, and main memory usage, respectively. In each subfigure, the results for different numbers of nodes ($n \in \{10, 25, 50, 81\}$) obtained with four methods are presented.

within 50 consecutive iterations, then $\lambda = \lambda/2$ and the Lagrangian multipliers will be set to the current best values. If the maximum iteration number is reached or the gap between the lower bound and upper bound is less than a certain value (≤0.1%) or all Lagrangian multipliers are equal to zero, then LR can be terminated. Note that the results for both GA and LR could be further improved by tuning the parameters. Since CBS provides a refinement of potential hubs, we use it with CPLEX together as a preselection method. In this experiment, its restricted variant RCBS is tested because of its short computation time compared with CBS and other variants. In addition, the results obtained by CPLEX are used as a benchmark.

GA and LR are implemented with *Python*, while the RCBS and LP are solved with CPLEX. Because CPLEX uses multiple threads as default, we use a single thread in all experiments to obtain comparable results. Finally, the maximum computation time is set to 3,600 seconds; that is, an experiment will be terminated once the running time exceeds 3,600 seconds. The results for the TR dataset, AP dataset, and CAB dataset are presented in Figures 5–8. The solution gaps, computation time, and main memory usage are used to measure the performances of four methods [65]. Because the observations on three datasets are generally similar, we only report the results for TR in detail here and summarize the differences to other datasets subsequently. All experiments were executed



(a) Convergence for the TR dataset

(b) Convergence for the AP dataset

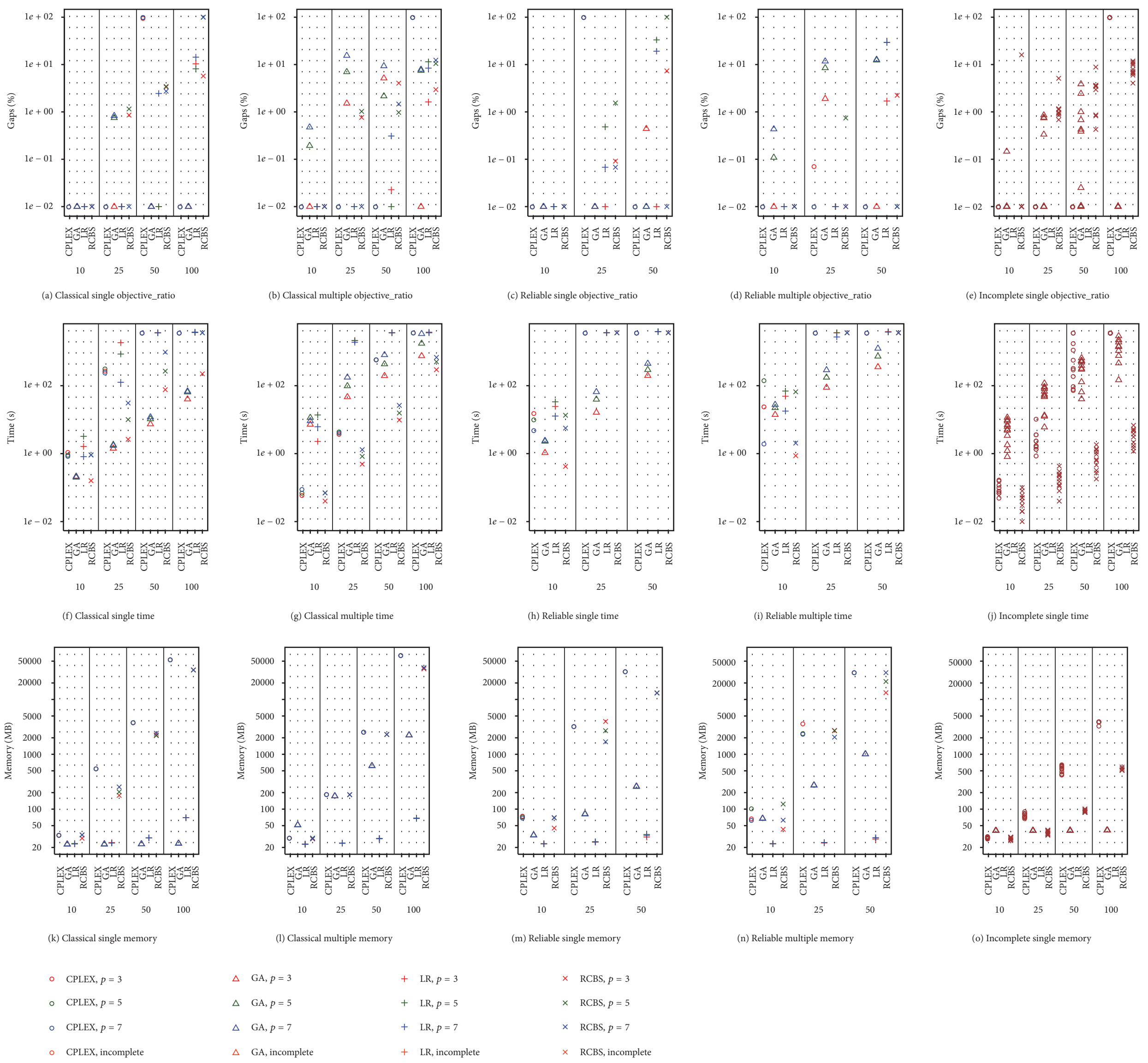
(c) Convergence for the CAB dataset

Figure 6: Convergence of CPLEX-LP and LR for RMA problems with $n = 25$, $p \in \{3, 5, 7\}$, and $\alpha \in \{0.3, 0.5, 0.7\}$ for the TR dataset and CAB dataset as well as $\alpha = 0.75$ for the AP dataset: LR converges gradually, whereas the gaps of CPLEX-LP stay almost stable after around 1,500 seconds.

on a server with 32 cores and 386 GB RAM, running Fedora 24 (Linux 4.8.4-200.fc24.x86_64).

4.2. Classical Single Allocation Problems. The solution gaps, computation time, and main memory usage of four methods for CSA problems in the TR dataset with different numbers of nodes ($n \in \{10, 25, 50, 81\}$) are shown in the first column (a, f, k) of Figure 5. Note that y-axis of all three figures is in log scale. In Figure 5(f), it can be seen that these methods all have similar computation time (around 1 second) while solving the problems with small size ($n = 10$). With an increasing number of nodes, the computation time for CLPEX-LP, LR, and RCBS increases significantly. In particular, when the number of nodes is up to 50, both CPLEX-LP and LR reach the maximum computation time of 3,600 seconds and they cannot provide good solutions in this case. However, GA performs well here. It provides very good solutions with small gaps within a reasonable computation time for different numbers of nodes. Note that RCBS can also terminate within 3,600 seconds for $n \leq 50$. However, it sometimes provides a poor solution with a large gap even for small number of nodes ($n \in \{10, 25\}$) because its additional constraints limit the search space too much. In Figure 5(k), the main memory usage of CPLEX-LP and RCBS increases significantly with an increasing number of nodes. When $n = 81$, the memory usage of these two CPLEX-based methods both is over 10 GB while GA and LR use much less memory than them. Therefore, GA performs the best for solving CSA problems in the TR dataset. In addition, it needs more computation time to obtain good solutions for larger numbers of hubs p with CPLEX-LP, GA, and RCBS.

4.3. Classical Multiple Allocation Problems. The computational complexity of multiple allocation problems is much lower than for single allocation problems. Therefore, CMA is solved in polynomial time if the hub set is determined [26], because the problem can be solved by finding the best route for each OD pair independently in this case. The results for CMA problems in the TR dataset are presented in the second column (b, g, l) of Figure 5. The results show that CPLEX-LP and RCBS take much shorter time for solving CMA than CSA. Thus, CPLEX-LP can be used to solve CMA problems within a reasonable time for small numbers of nodes ($n \leq 50$). However, just like the performance in CSA



(a) Classical single objective_ratio

(b) Classical multiple objective_ratio

(c) Reliable single objective_ratio

(d) Reliable multiple objective_ratio

(e) Incomplete single objective_ratio

(f) Classical single time

(g) Classical multiple time

(h) Reliable single time

(i) Reliable multiple time

(j) Incomplete single time

(k) Classical single memory

(l) Classical multiple memory

(m) Reliable single memory

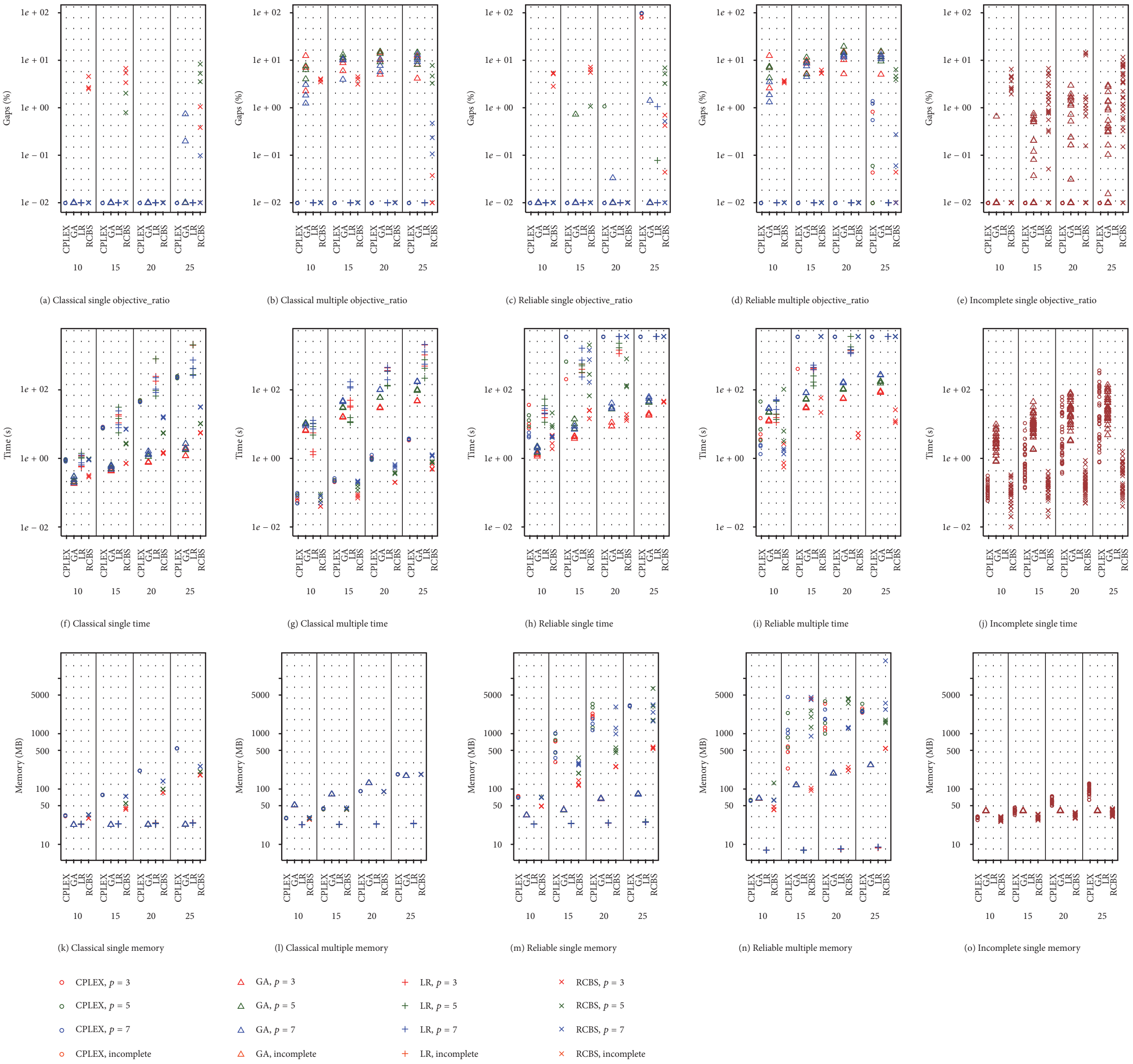
(n) Reliable multiple memory

(o) Incomplete single memory

FIGURE 7: Results of five types of problems for AP dataset. The problems in five columns are CSA, CMA, RSA, RMA, and ISA. The measurements of three rows are solution gaps, computation time, and main memory usage, respectively. In each subfigure, the results for different numbers of nodes ($n \in \{10, 25, 50, 100\}$) obtained with four methods are presented.

problems, RCBS sometimes provides relatively poor solutions with large gaps. Note that GA becomes the worst method with large-gap solutions and long computation time because in the case of multiple allocation, the size of chromosomes is up to the number of OD pairs ($O(n^2)$). It is more difficult to reach the near-optimal solutions with random exploration in this case. We have increased the population size to 1000 in additional experiments with GA; however, results did not improve significantly. LR still needs a long computation time for large number of nodes. In Figure 5(l), similar to Figure 5(i), CPLEX-LP and RCBS need much more memory with an increasing number of nodes, while the memory usage of LR is always much less than them. Because of the larger size of chromosomes, GA uses more memory for CMA than CSA problems. Finally, similar to Section 4.2, it is more difficult to obtain good solutions for larger p with CPLEX-LP, GA, and RCBS.

4.4. Reliable Single Allocation Problems. The reliable problems have more computation complexity than the classical problems. Thus, the maximum number of nodes for the evaluation of two reliable problems is set to $n = 50$. The performance of four approaches for solving RSA problems is shown in the third column (c, h, m) of Figure 5, indicating that when the number of nodes is $n \geq 25$, we cannot obtain any optimal solution with CPLEX-LP directly. The results provided by RCBS are slightly better, but it cannot still guarantee the optimality of the solutions even for networks with small sizes. LR can provide acceptable solutions in networks of small size ($n = 10$). However, with an increasing



(a) Classical single objective_ratio
(b) Classical multiple objective_ratio
(c) Reliable single objective_ratio
(d) Reliable multiple objective_ratio
(e) Incomplete single objective_ratio
(f) Classical single time
(g) Classical multiple time
(h) Reliable single time
(i) Reliable multiple time
(j) Incomplete single time
(k) Classical single memory
(l) Classical multiple memory
(m) Reliable single memory
(n) Reliable multiple memory
(o) Incomplete single memory

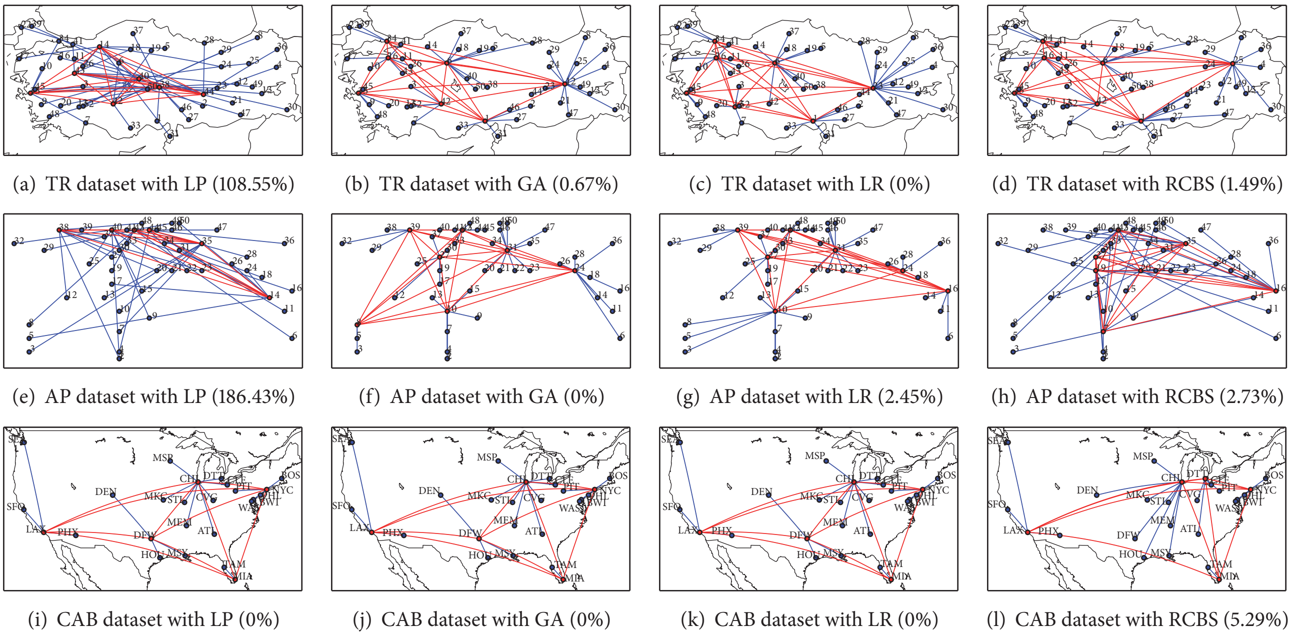
Figure 8: Results of five types of problems for CAB dataset. The problems in five columns are CSA, CMA, RSA, RMA, and ISA. The measurements of three rows are solution gaps, computation time, and main memory usage, respectively. In each subfigure, the results for different numbers of nodes ($n \in \{10, 15, 20, 25\}$) obtained with four methods are presented.

number of nodes, the solution gaps become large because of the slow computation for each iteration. The observation in Figure 5(m) is still similar to Figures 5(k) and 5(l): the main memory usage of CPLEX-LP and RCBS increases fast with increasing numbers of nodes, while GA and LR use much less memory than them. Finally, similar to the case of CSA problems, GA presents solutions with high quality within a reasonable time.

4.5. Reliable Multiple Allocation Problems. The results for the RMA problems are presented in the fourth column (d, i, n) of Figure 5. Similar to the condition in RSA problems, the optimal solutions cannot be obtained with CPLEX-LP within 3,600 seconds for $n \geq 25$. However, the gaps are smaller than in Figure 5(c), because of the lower complexity of multiple allocation problems. GA still performs the worst and RCBS cannot still guarantee the optimality always. These observations are similar to Section 4.3. The influence of the number of hubs p is still similar to the sections above.

In general, RMA problems with $n = 10$ can be solved by CPLEX-LP or LR with small gaps (<0.1%) within 3,600 seconds. However, for the cases of $n \geq 25$, the problem cannot be solved within 3,600 seconds by any methods proposed in this paper. Therefore, an additional evaluation for the convergence is performed below.

The convergence progress of CPLEX-LP and LR for RMA problems with $n = 25$ within 3,600 seconds is shown in Figure 6 which indicates that with the increase of time the



(a) TR dataset with LP (108.55%)

(b) TR dataset with GA (0.67%)

(c) TR dataset with LR (0%)

(d) TR dataset with RCBS (1.49%)

(e) AP dataset with LP (186.43%)

(f) AP dataset with GA (0%)

(g) AP dataset with LR (2.45%)

(h) AP dataset with RCBS (2.73%)

(i) CAB dataset with LP (0%)

(j) CAB dataset with GA (0%)

(k) CAB dataset with LR (0%)

(l) CAB dataset with RCBS (5.29%)

FIGURE 9: Visualization of CSA problems for the TR dataset with $n = 50$, $p = 7$, and $\alpha = 0.5$, AP dataset with $n = 50$, $p = 7$, and $\alpha = 0.75$, and CAB dataset with $n = 25$, $p = 5$, and $\alpha = 0.5$: red points are hubs and blue points are nonhub nodes. The red links are hub links. For the TR dataset, CPLEX-LP provides a feasible solution only, and therefore the assignment is rather useless. The results obtained by GA, LR, and RCBS are quite similar to each other. For the AP dataset, CPLEX-LP and RCBS provide poor solutions, while the results of GA and LR are different regarding one hub. For the CAB dataset, CPLEX-LP, GA, and LR are all up to the optimality, while RCBS selects Detroit (DTT) as a hub because of its high node importance instead of Dallas (DFW).

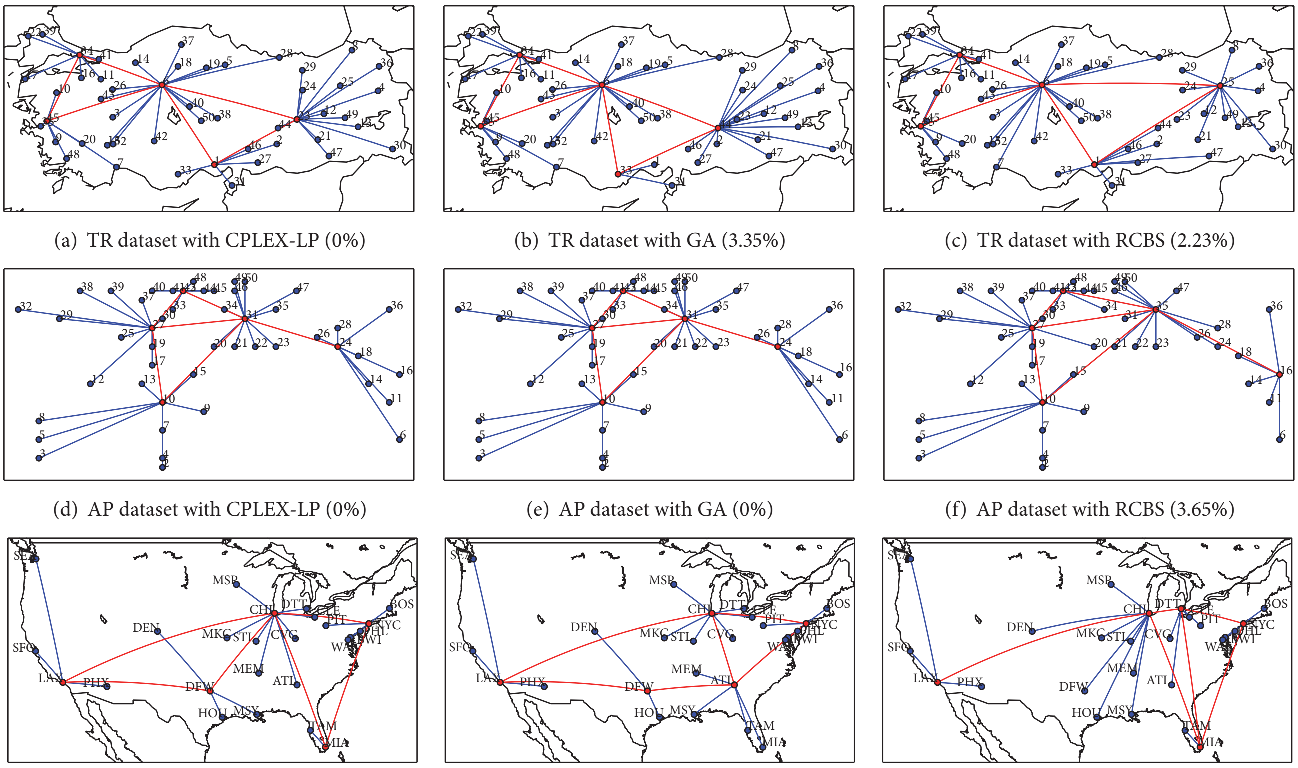
solutions obtained by LR converge gradually. Finally, the gaps approach small numbers. However, although CPLEX-LP can provide gaps less than 10% within 1,500 seconds for all 9 cases ($p \in \{3, 5, 7\}, \alpha \in \{0.3, 0.5, 0.7\}$), the gaps stay nearly stable within the remaining computation time; that is, providing more computational resources does not improve the solution qualities. In addition, GA and RCBS cannot provide a near-optimal solution for RMA because of the algorithm structure. Therefore, LR is the best technique to solve reliable multiple allocation problems.

4.6. Incomplete Single Allocation Problems. For the ISA problems, the number of hub links q also needs to be considered in addition to the number of hub p. We evaluate the following combinations of p and q: $(p, q) \in \{(2, 1), (3, 2), (3, 3), (4, 4), (4, 5), (4, 6), (5, 6), (5, 7), (5, 8), (5, 9), (5, 10)\}$, similar to existing work on ISA problems [18]. The results are shown in the fifth column (e, j, o) of Figure 5. We only evaluate CPLEX-LP, GA, and RCBS here, because LR is a nontrivial algorithm and has not yet been published for solving ISA problems. CPLEX-LP still performs well in small networks ($n \leq 25$) and RCBS cannot guarantee the optimality. GA strategy proposed by us performs worse than CPLEX-LP for $n \leq 25$. However, it can guarantee small gaps ($\leq$3%) for all the instances while CPLEX-LP and RCBS often provide gaps over 10%. The scale of variables of ISA problems is $O(n^3)$ in (37)–(48). Thus, the main memory usage of these three methods for ISA problems all is much less than that for the other four problems.

4.7. Results for the AP Dataset and CAB Dataset. In addition to the TR dataset, the results for the AP dataset and CAB dataset are presented in Figures 7 and 8. The results corroborate our findings for the TR dataset proposed in Sections 4.2 to 4.6. CPLEX-LP computes optimal solutions for CSA, CMA, and ISA problems in small instances because of their low complexity. GA still has the best performance, with short computation time and low solution gaps for solving CSA, RSA, and ISA problems, because the problem formulations can be neatly encoded with chromosomes of reasonable sizes. LR needs a long computation time for all problems with large network scales. However, similar to the results in Section 4.5, RMA with large networks scales cannot be solved by any of these methods within 3,600 seconds. From the observation in Figure 6, LR should be used to solve RMA problems. The computation time of RCBS is shorter than LP. However, RCBS sometimes provides poor solutions with high gaps because of the additional constraints in the method.

4.8. Visualization of Results. In the sections above, we analyzed the solution gaps, computation time, and main memory usage of four solution techniques for five types of hub location problems. In this section, the results of CSA and ISA problems for the three datasets are visualized in Figures 9 and 10. The number at the end of the caption of each subfigure is the relative difference between the solution and the minimum one.

In the first row (a–d) of Figure 9, results of CSA problems for the TR dataset with $n = 50$, $p = 7$, $\alpha = 0.5$ are proposed.



(a) TR dataset with CPLEX-LP (0%)

(b) TR dataset with GA (3.35%)

(c) TR dataset with RCBS (2.23%)

(d) AP dataset with CPLEX-LP (0%)

(e) AP dataset with GA (0%)

(f) AP dataset with RCBS (3.65%)

(g) CAB dataset with CPLEX-LP (0%)

(h) CAB dataset with GA (1.33%)

(i) CAB dataset with RCBS (4.37%)

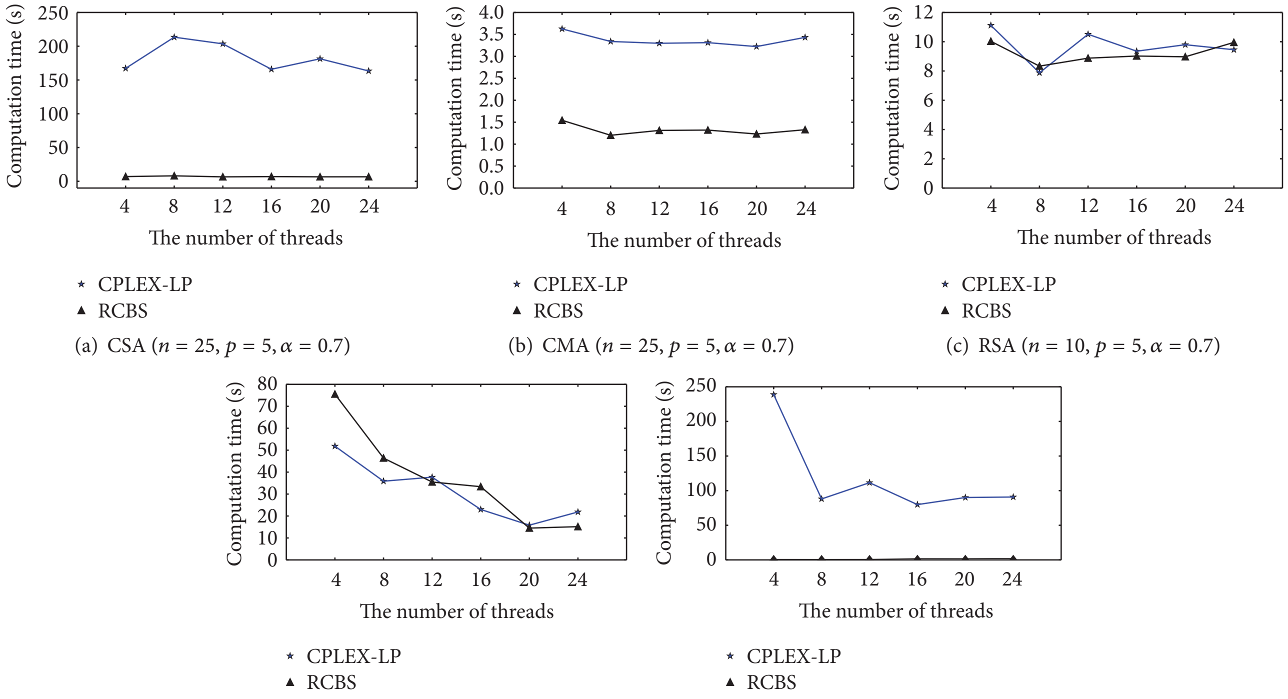
Figure 10: Visualization of ISA problems for the TR dataset with $n = 50$, $p = 5$, $q = 6$, and $\alpha = 0.5$, AP dataset with $n = 50$, $p = 5$, $q = 6$, and $\alpha = 0.75$, and CAB dataset with $n = 25$, $p = 5$, $q = 6$, and $\alpha = 0.5$: red points and blue points are hubs and nonhub nodes. The red links are the links between hubs. For the TR dataset, the structures of links obtained by three methods are similar to each other. For the AP dataset, the optimal solution is obtained with CPLEX-LP and GA while RCBS selects node 35 and node 16 as hubs because of their high importance. For the CAB dataset, CPLEX-LP provides an optimal solution. The solutions of GA and CPLEX-LP only have a difference of 1.33% while having significantly different hub assignments in the figures. RCBS still selects Detroit (DTT) as a hub instead of Dallas (DFW).

CPLEX-LP only provides a feasible solution in 3,600 seconds because of the large network scale. Thus, the assignment in Figure 9(a) is rather useless. The assignments of other three methods also have some difference, and their solutions are not far away from each other. The results of CSA problems for AP dataset with $n = 50$, $p = 7$, $\alpha = 0.75$ are shown in the second row (e–h) of Figure 9. CPLEX-LP and RCBS provide only feasible solutions with bad assignments. The results of GA and LR have a difference on one hub, but the difference between the values of their objective functions is only 2.45%. In the third row (i–l) of Figure 9, results of CSA problems for the CAB dataset with $n = 25$, $p = 5$, $\alpha = 0.5$ are shown. It indicates that the hub sets and node assignments obtained by CPLEX-LP, GA, and LR are all optimal. However, the results of RCBS are different: Detroit (DTT) is selected as a hub because of its high node importance (obtained by (57)) instead of Dallas (DFW).

In the first row (a–c) of Figure 10, results of ISA problems for the TR dataset with $n = 50, p = 5, q = 6, \alpha = 0.5$ are presented. The structures of links obtained by three methods are similar to each other. Only a few of hubs are different between three figures. In the second row (d–f) of Figure 10, results for the AP dataset with $n = 50$, $p = 5, q = 6, \alpha = 0.75$ are shown. The optimal solution is obtained with CPLEX-LP and GA. RCBS selects node 35 and node 16 as hubs because of their high importance. Results for the CAB dataset with $n = 25, p = 5, q = 6, \alpha = 0.5$ with three methods are shown in the third row (g–i) of Figure 10. Here, the optimal solution is obtained with CPLEX-LP. Although the assignment provided by GA looks quite different from that by CPLEX-LP, their solutions only have a difference of 1.33%. Detroit (DTT) is still selected as a hub instead of Dallas (DFW) by RCBS.

4.9. Analysis of Parallel Computation. The results from Sections 4.2 to 4.7 are obtained using a single thread only. However, the number of threads can be modified easily for CPLEX-LP and RCBS. In this section, we analyze the performance of these two solution techniques with different numbers of threads.

As shown in Figure 11, two classical problems with $n = 25$, $p = 5$, $\alpha = 0.7$, two reliable problems with $n = 10$, $p = 5$, $\alpha = 0.7$, and the ISA problem with $n = 25$, $p = 5$, $q = 6$, $\alpha = 0.7$ for the CAB dataset are used as case studies. CPLEX-LP and RCBS provide good solutions, with gaps less than 0.1% within 3,600 seconds for all five problems. Thus, we consider the computation time in this section. It can be seen that in the RMA problem (Figure 11(d)) the computation time of two solution techniques can be reduced



(a) CSA ($n = 25, p = 5, \alpha = 0.7$)

(b) CMA ($n = 25, p = 5, \alpha = 0.7$)

(c) RSA ($n = 10, p = 5, \alpha = 0.7$)

(d) RMA ($n = 10, p = 5, \alpha = 0.7$)

(e) ISA ($n = 25, p = 5, q = 9, \alpha = 0.7$)

FIGURE 11: Computation time of LP and RCBS with different number of threads for the CAB dataset: for the RMA problem, the computation time of two solution techniques can be reduced significantly with an increasing number of threads. For the ISA problem, the computation time of CPLEX-LP can be reduced by adding the threads when the number of threads is less than a fixed number (here, the fixed number is 8).

significantly with an increasing number of threads. For the ISA problem (Figure 11(e)), the computation time can be reduced by adding the threads when the number of threads is less than a fixed number (here, the fixed number is 8). However, in the other three figures, there is no significant correlation between computation time and the number of threads. For future work, the parallel computation of GA and LR should be implemented. The impact of the number of threads will also be analyzed.

5. Conclusions

In this paper, we reviewed, implemented, and compared several methods (genetic algorithm, Lagrangian relaxation, clustering-based method, and CPLEX) for solving five types of hub location problems (CSA, CMA, RSA, RMA, and ISA). Because most methods were proposed for solving one or two types of hub location problems, we made appropriate modifications for these methods to solve all kinds of hub location problems. In order to evaluate the scalability and solution quality of these methods under the same computational environment, three standard datasets (TR, AP, and CAB) were used as case studies. In the evaluation, gaps of the objective functions, computation time, and main memory usage were used to compare the performance of these methods. There are seven major observations obtained from our evaluation:

(1) CPLEX-LP performs best on solving two classical (CSA, CMA) problems and incomplete single allocation (ISA) problems with small size because of the low problem complexity.

(2) GA provides good solutions within short time for solving three single allocation (CSA, RSA, and ISA) problems, because small chromosomes are sufficient for modeling single allocation problems. However, it might be a poor choice for solving multiple allocation (CMA and RMA) problems, because the size of the chromosomes is in $O(n^2)$ (here n is the number of nodes).

(3) LR should be used to solve reliable multiple allocation (RMA) problems, since other methods perform poorly for solving these problems, because the computational complexity of RMA and CMA is similar for LR. However, while using other methods, the complexity of RMA is much higher than CMA.

(4) RCBS can be used to obtain a solution within a shorter time than CPLEX-LP. However, because there are additional constraints in the algorithm, the optimality of the solution cannot be guaranteed with RCBS. Although these constraints can reduce the computation time, the problems may be changed already. Because of the constraints of potential hub set, some nodes with high importance are more likely to be

selected as hubs no matter whether they are the optimal hubs, such as Detroit (DTT) in Figures 9(l) and 10(i).

(5) CPLEX-LP, GA, and RCBS use more main memory for larger networks, whereas the memory usage of LR remains almost stable for different numbers of nodes. CPLEX-LP and RCBS need much more memory than GA and LR.

(6) The structures of solutions obtained with several methods are sometimes completely different, yet, the values of their objective functions usually are not so far away from each other. For a given problem, more computation time is needed to obtain good solutions for large numbers of hubs p with CPLEX-LP, GA, and RCBS.

(7) For reliable multiple allocation (RMA) problems, the computation time of CPLEX-LP and RCBS can be reduced significantly with an increasing number of threads. For incomplete single allocation (ISA) problems, the computation time of CPLEX-LP can be reduced by adding the threads when the number of threads is less than a fixed number. However, there is no significant correlation between computation time and the number of threads for other three types of problems.

Finally, note that several other types of hub location problems, such as the uncapacitated hub location problem and the r-allocation p-hub median problem [66], are not evaluated in this paper. For further work, these different problems and solution techniques should be considered. Hub location problems can be applied to other network design problems, multimodal systems, or multiple airport systems [67–70], and this can also be studied for future work. In this paper, we used CPLEX directly; future work could implement more sophisticated methods, such as Benders decomposition. In addition, GA performs well in single allocation problems in the current study. However, the computation time and the usage of main memory increase quickly with the growth of the network. Therefore, the combination of GA and CBS can be considered, that is, operating the GA evolution in the initial population obtained by CBS. In addition, the compression for representation of allocation arrays can reduce the usage of main memory. The parallel computation of several independent subpopulations in multiple threads may also speed up the algorithms.

Abbreviation Summary of Methods

AP:	Australia Post
BB:	Branch-and-bound
BP:	Branch-and-price
(R)CBS:	(Restricted) clustering-based potential hub set
EBD:	Enhanced Benders decomposition
GA:	Genetic algorithm
LAA:	Location-allocation algorithm
LP:	Linear program
LR:	Lagrangian relaxation
PRM:	Path relinking method
RCA:	Relax-and-cut algorithm
SAA:	Simulated annealing algorithm
TS:	Tabu search
WA:	Weiszfeld algorithm.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This study is supported by the research fund from National Natural Science Foundation of China (Grant nos. 61601013, 61650110516, 61521091, and 91538204).

References

[1] M. E. O’elly, “The location of interacting hub facilities,” *Transportation science*, vol. 20, no. 2, pp. 92–106, 1986.

[2] S. Gelareh and S. Nickel, “Hub location problems in transportation networks,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 47, no. 6, pp. 1092–1111, 2011.

[3] Y. An, B. Zeng, Y. Zhang, and L. Zhao, “Reliable p-median facility location problem: Two-stage robust models and algorithms,” *Transportation Research Part B: Methodological*, vol. 64, pp. 54–72, 2014.

[4] S. Gelareh, R. Neamatian Monemi, and S. Nickel, “Multiperiod hub location problems in transportation,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 75, pp. 67–94, 2015.

[5] S. Wang, X. Qu, T. Wang, and W. Yi, “Optimal Container Routing in Liner Shipping Networks Considering Repacking 20 ft Containers into 40 ft Containers,” *Journal of Advanced Transportation*, vol. 2017, 9 pages, 2017.

[6] Y. Park and M. E. O’Kelly, “Origin-destination synthesis for aviation network data: examining hub operations in the domestic and international US markets,” *Journal of Advanced Transportation*, vol. 50, no. 8, pp. 2288–2305, 2016.

[7] H. Yaman and G. Carello, “Solving the hub location problem with modular link capacities,” *Computers & Operations Research*, vol. 32, no. 12, pp. 3227–3245, 2005.

[8] H. Kim and M. E. O’Kelly, “Reliable p-hub location problems in telecommunication networks,” *Geographical Analysis*, vol. 41, no. 3, pp. 283–306, 2009.

[9] S. Alumur and B. Y. Kara, “Network hub location problems: the state of the art,” *European Journal of Operational Research*, vol. 190, no. 1, pp. 1–21, 2008.

[10] J. F. Campbell and M. E. O’Kelly, “Twenty-five years of hub location research,” *Transportation Science*, vol. 46, no. 2, pp. 153–169, 2012.

[11] R. Z. Farahani, M. Hekmatfar, A. B. Arabani, and E. Nikbakhsh, “Hub location problems: a review of models, classification, solution techniques, and applications,” *Computers & Industrial Engineering*, vol. 64, no. 4, pp. 1096–1109, 2013.

[12] M. E. O’Kelly, “A quadratic integer program for the location of interacting hub facilities,” *European Journal of Operational Research*, vol. 32, no. 3, pp. 393–404, 1987.

[13] J. F. Campbell, “Hub location and the p-hub median problem,” *Operations Research*, vol. 44, no. 6, pp. 923–935, 1996.

[14] S. Wandelt, X. Sun, and X. Cao, “Computationally efficient attack design for robustness analysis of air transportation networks,” *Transportmetrica A: Transport Science*, vol. 11, no. 10, pp. 939–966, 2015.

[15] S. Wandelt, X. Sun, M. Zanin, and S. Havlin, “QRE: quick robustness estimation for large complex networks,” *Future Generation Computer Systems*, 2017.

[16] X. Sun, V. Gollnick, and S. Wandelt, “Robustness analysis metrics for worldwide airport network: a comprehensive study,” *Chinese Journal of Aeronautics*, vol. 30, no. 2, pp. 500–512, 2017.

[17] X. Sun, S. Wandelt, and F. Linke, “On the topology of air navigation route systems,” *Proceedings of the ICE - Transport*, vol. 170, no. 1, pp. 46–59, 2017.

[18] S. A. Alumur, B. Y. Kara, and O. E. Karasan, “The design of single allocation incomplete hub networks,” *Transportation Research Part B: Methodological*, vol. 43, no. 10, pp. 936–951, 2009.

[19] T. Meyer, A. T. Ernst, and M. Krishnamoorthy, “A 2-phase algorithm for solving the single allocation p-hub center problem,” *Computers & Operations Research*, vol. 36, no. 12, pp. 3143–3151, 2009.

[20] C. Ortiz-Astorquiza, I. Contreras, and G. Laporte, “Multi-level facility location as the maximization of a submodular set function,” *European Journal of Operational Research*, vol. 247, no. 3, Article ID 13006, pp. 1013–1016, 2015.

[21] D. Skorin-Kapov and J. Skorin-Kapov, “On tabu search for the location of interacting hub facilities,” *European Journal of Operational Research*, vol. 73, no. 3, pp. 502–509, 1994.

[22] A. T. Ernst and M. Krishnamoorthy, “Efficient algorithms for the uncapacitated single allocationp-hub median problem,” *Location Science*, vol. 4, no. 3, pp. 139–154, 1996.

[23] H. Pirkul and D. A. Schilling, “An efficient procedure for designing single allocation hub and spoke systems,” *Management Science*, vol. 44, no. 12, pp. S235–S242, 1998.

[24] I. Contreras, J. A. Daz, and E. Fernández, “Branch and price for large-scale capacitated hub location problems with single assignment,” *INFORMS Journal on Computing*, vol. 23, no. 1, pp. 41–55, 2011.

[25] S. Elhedhli and H. Wu, “A Lagrangean heuristic for hub-and-spoke system design with capacity selection and congestion,” *INFORMS Journal on Computing*, vol. 22, no. 2, pp. 282–296, 2010.

[26] Y. An, Y. Zhang, and B. Zeng, “The reliable hub-and-spoke design problem: Models and algorithms,” *Transportation Research Part B: Methodological*, vol. 77, pp. 103–122, 2015.

[27] H. Topcuoglu, F. Corut, M. Ermis, and G. Yilmaz, “Solving the uncapacitated hub location problem using genetic algorithms,” *Computers & Operations Research*, vol. 32, no. 4, pp. 967–984, 2005.

[28] J. Kratica, Z. Stanimirović, D. s. Tošić, and V. Filipović, “Two genetic algorithms for solving the uncapacitated single allocation p-hub median problem,” *European Journal of Operational Research*, vol. 182, no. 1, pp. 15–28, 2007.

[29] I. Contreras, J.-F. Cordeau, and G. Laporte, “Benders decomposition for large-scale uncapacitated hub location,” *Operations Research*, vol. 59, no. 6, pp. 1477–1490, 2011.

[30] Z. Stanimirović, “A genetic algorithm approach for the capacitated single allocation p-hub median problem,” *Computing and Informatics*, vol. 29, no. 1, pp. 117–132, 2012.

[31] H. Damgacioglu, D. Dinler, N. E. Evin Ozdemirel, and C. Iyigun, “A genetic algorithm for the uncapacitated single allocation planar hub location problem,” *Computers & Operations Research*, vol. 62, pp. 224–236, 2015.

[32] M. Peker, B. Y. Kara, J. F. Campbell, and S. A. Alumur, “Spatial analysis of single allocation hub location problems,” *Networks and Spatial Economics*, vol. 16, no. 4, pp. 1075–1101, 2016.

[33] N. Azizi, S. Chauhan, S. Salhi, and N. Vidyarthi, “The impact of hub failure in hub-and-spoke networks: Mathematical formulations and solution techniques,” *Computers & Operations Research*, vol. 65, pp. 174–188, 2016.

[34] R. S. de Camargo, G. de Miranda, M. E. O’Kelly, and J. F. Campbell, “Formulations and decomposition methods for the incomplete hub location network design problem with and without hop-constraints,” *Applied Mathematical Modelling*, vol. 51, C, pp. 274–301, 2017.

[35] M. P. Pérez, F. A. Rodrguez, and J. M. M. Vega, “On the use of path relinking for the ρ–hub median problem,” in *European Conference on Evolutionary Computation in Combinatorial Optimization*, pp. 155–164, Springer, 2004.

[36] M. Bashiri, M. Mirzaei, and M. Randall, “Modeling fuzzy capacitated p-hub center problem and a genetic algorithm solution,” *Applied Mathematical Modelling*, vol. 37, no. 5, pp. 3513–3525, 2013.

[37] J. Zheng, Q. Meng, and Z. Sun, “Liner hub-and-spoke shipping network design,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 75, pp. 32–48, 2015.

[38] E. Weiszfeld, “Sur le point pour lequel la somme des distances de n points donnés est minimum,” *Tohoku Mathematical Journal*, vol. 43, no. 2, pp. 355–386, 1937.

[39] T. Aykin and G. F. Brown, “Interacting new facilities and location-allocation problems,” *Transportation Science*, vol. 26, no. 3, pp. 212–222, 1992.

[40] T. Cui, Y. Ouyang, and Z. M. Shen, “Reliable facility location design under the risk of disruptions,” *Operations Research*, vol. 58, no. 4, part 1, pp. 998–1011, 2010.

[41] M. Iwasa, H. Saito, and T. Matsui, “Approximation algorithms for the single allocation problem in hub-and-spoke networks and related metric labeling problems,” *Discrete Applied Mathematics*, vol. 157, no. 9, pp. 2078–2088, 2009.

[42] I. Correia, S. Nickel, and F. Saldanha-da-Gama, “Single-assignment hub location problems with multiple capacity levels,” *Transportation Research Part B: Methodological*, vol. 44, no. 8-9, pp. 1047–1066, 2010.

[43] R. S. de Camargo, G. d. Miranda, and H. Luna, “Benders decomposition for the uncapacitated multiple allocation hub location problem,” *Computers Operations Research*, vol. 35, no. 4, pp. 1047–1064, 2008.

[44] L. Cánovas, S. Garca, and A. Marn, “Solving the uncapacitated multiple allocation hub location problem by means of a dual-ascent technique,” *European Journal of Operational Research*, vol. 179, no. 3, pp. 990–1007, 2007.

[45] A. Marín, L. Cánovas, and M. Landete, “New formulations for the uncapacitated multiple allocation hub location problem,” *European Journal of Operational Research*, vol. 172, no. 1, pp. 274–292, 2006.

[46] J. Puerto, A. B. Ramos, A. M. Rodríguez-Chía, and M. C. Sánchez-Gil, “Ordered median hub location problems with capacity constraints,” *Transportation Research Part C: Emerging Technologies*, vol. 70, pp. 142–156, 2016.

[47] Q. Meng and X. Wang, "Intermodal hub-and-spoke network design: incorporating multiple stakeholders and multi-type containers," *Transportation Research Part B: Methodological*, vol. 45, no. 4, pp. 724–742, 2011.

[48] S. A. Alumur, S. Nickel, and F. Saldanha-da-Gama, "Hub location under uncertainty," *Transportation Research Part B: Methodological*, vol. 46, no. 4, pp. 529–543, 2012.

[49] S. A. Alumur, H. Yaman, and B. Y. Kara, "Hierarchical multimodal hub location problem with time-definite deliveries," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 6, pp. 1107–1120, 2012.

[50] A. I. Mahmutogullari and B. Y. Kara, "Hub location under competition," *European Journal of Operational Research*, vol. 250, no. 1, pp. 214–225, 2016.

[51] S. Çetiner, *An iterative hub location and routing problem for postal delivery systems [Ph.D. thesis]*, Citeseer, 2003.

[52] J.-F. Chen, "A hybrid heuristic for the uncapacitated single allocation hub location problem," *Omega* , vol. 35, no. 2, pp. 211–220, 2006.

[53] S. Belkoura, J. M. Peña, and M. Zanin, "Beyond linear delay multipliers in air transport," *Journal of Advanced Transportation*, vol. 2017, 11 pages, 2017.

[54] L. V. Snyder and M. S. Daskin, "Reliability models for facility location: the expected failure cost case," *Transportation Science*, vol. 39, no. 3, pp. 400–416, 2005.

[55] J. F. Meier and U. Clausen, "Solving classical and new single allocation hub location problems on euclidean data," Tech. Rep., Optimization Online, 2015, 2015, http://www.optimization-online.org/DB_HTML/2015/03/4816.html.

[56] I. Contreras, J.-F. Cordeau, and G. Laporte, "Stochastic uncapacitated hub location," *European Journal of Operational Research*, vol. 212, no. 3, pp. 518–528, 2011.

[57] D. Zhang, F. Zou, S. Li, and L. Zhou, "Green Supply Chain Network Design with Economies of Scale and Environmental Concerns," *Journal of Advanced Transportation*, vol. 2017, 14 pages, 2017.

[58] J. G. Klincewicz, "A dual algorithm for the uncapacitated hub location problem," *Location Science*, vol. 4, no. 3, pp. 173–184, 1996.

[59] J. F. Campbell, "Integer programming formulations of discrete hub location problems," *European Journal of Operational Research*, vol. 72, no. 2, pp. 387–405, 1994.

[60] J. Ebery, M. Krishnamoorthy, A. Ernst, and N. Boland, "Capacitated multiple allocation hub location problem: Formulations and algorithms," *European Journal of Operational Research*, vol. 120, no. 3, pp. 614–631, 2000.

[61] W. Chaovalitwongse, P. M. Pardalos, and O. A. Prokopyev, "A new linearization technique for multi-quadratic 0-1 programming problems," *Operations Research Letters*, vol. 32, no. 6, pp. 517–522, 2004.

[62] C. B. Cunha and M. R. Silva, "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*, vol. 179, no. 3, pp. 747–758, 2007.

[63] R. C. Prim, "Shortest connection networks and some generalizations," *Bell Labs Technical Journal*, vol. 36, no. 6, pp. 1389–1401, 1957.

[64] M. L. Fisher, "The lagrangian relaxation method for solving integer programming problems," *Management Science*, vol. 50, no. 12, pp. 1861–1874, 2004.

[65] M. Drexl and M. Schneider, "A survey of variants and extensions of the location-routing problem," *European Journal of Operational Research*, vol. 241, no. 2, pp. 283–308, 2015.

[66] H. Yaman, "Allocation strategies in hub networks," *European Journal of Operational Research*, vol. 211, no. 3, pp. 442–451, 2011.

[67] C. Li, H. Yang, D. Zhu, and Q. Meng, "A global optimization method for continuous network design problems," *Transportation Research Part B: Methodological*, vol. 46, no. 9, pp. 1144–1158, 2012.

[68] S. Wang, Q. Meng, and H. Yang, "Global optimization methods for the discrete network design problem," *Transportation Research Part B: Methodological*, vol. 50, pp. 42–60, 2013.

[69] X. Sun, S. Wandelt, M. Hansen, and A. Li, "Multiple airport regions based on inter-airport temporal distances," *Transportation Research Part E: Logistics and Transportation Review*, vol. 101, pp. 84–98, 2017.

[70] X. Sun, Y. Zhang, and S. Wandelt, "Air transport versus high-speed rail: an overview and research agenda," *Journal of Advanced Transportation*, vol. 2017, Article ID 8426926, 18 pages, 2017.

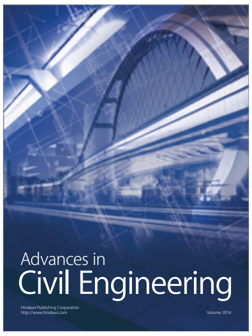
Journal of
Engineering

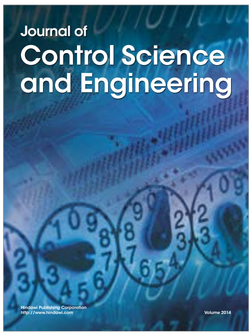
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

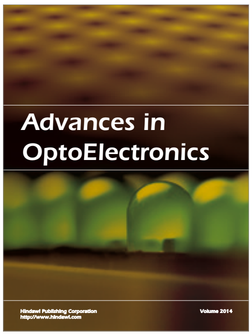
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

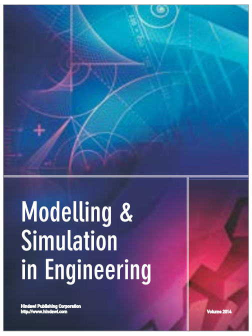
Journal of
Robotics

Journal of
Electrical and Computer Engineering

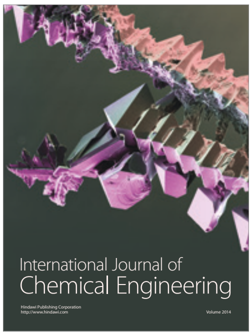

Advances in
OptoElectronics

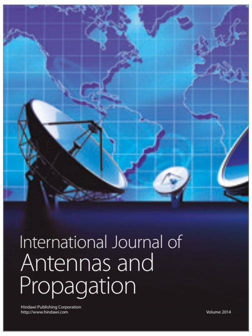
VLSI Design

International Journal of
Navigation and Observation

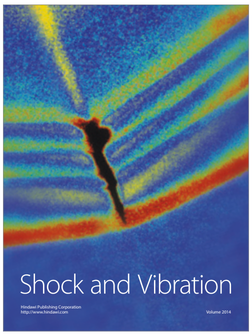

Modelling &
Simulation
in Engineering

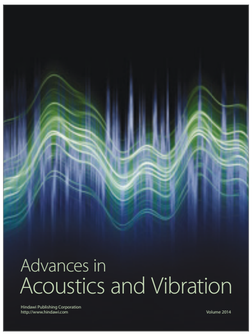
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration