Health Care Manag Sci
DOI 10.1007/s10729-015-9332-4

# Iterative optimization algorithm with parameter estimation for the ambulance location problem

Sun Hoon Kim[1] · Young Hoon Lee[1]

Received: 25 February 2015 / Accepted: 19 June 2015


**Abstract** The emergency vehicle location problem to determine the number of ambulance vehicles and their locations satisfying a required reliability level is investigated in this study. This is a complex nonlinear issue involving critical decision making that has inherent stochastic characteristics. This paper studies an iterative optimization algorithm with parameter estimation to solve the emergency vehicle location problem. In the suggested algorithm, a linear model determines the locations of ambulances, while a hypercube simulation is used to estimate and provide parameters regarding ambulance locations. First, we suggest an iterative hypercube optimization algorithm in which interaction parameters and rules for the hypercube and optimization are identified. The interaction rules employed in this study enable our algorithm to always find the locations of ambulances satisfying the reliability requirement. We also propose an iterative simulation optimization algorithm in which the hypercube method is replaced by a simulation, to achieve computational efficiency. The computational experiments show that the iterative simulation optimization algorithm performs equivalently to the iterative hypercube optimization. The suggested algorithms are found to outperform existing algorithms suggested in the literature.

**Keywords** Optimization · Iterative Approach · Hypercube · Simulation · Reliability Level

✉ Sun Hoon Kim
misia789@hanmail.net

1 Department of information and Industrial Engineering, Yonsei University, 50 Yonsei-Ro, Seodaemun-Gu, Seoul 120-749, Korea

## 1 Introduction

In an emergency medical service system (EMS), the ambulance location problem (ALP) is a critical issue. Locating ambulances is crucial to providing timely emergency medical services, affecting patients' lives intimately. The ALP can be defined as the problem to find the number and locations of ambulances needed to provide a certain level of timely service. The location set cover problem (LSCP), which is to minimize the number of vehicles required to cover all demand sites within a specific distance, is an effective approach used to model the ALP. However, the LSCP is not able to capture the most important characteristic of the ALP: unavailability of an ambulance, such as when it is occupied by a patient, which leads to the late arrival of the ambulance at the target location, thus decreasing the patient's chances of survival. Therefore, it is important to build a reliable ambulance location model.

In 1980s, the term *reliability* emerged regarding the ALP to address the issue of unavailability of ambulances. It can be defined as the probability that a patient who calls for an ambulance will be provided with one: reliability=(total calls—lost calls)/total calls. Thus, to ensure the reliability of demand sites, ambulance unavailability must be controlled in the ALP. ReVelle and Horgan [1–3] suggested the probabilistic LSCP (PLSCP) in which a reliability factor was embodied as a constraint. This problem seeks to find appropriate locations for the ambulances to ensure that the reliability levels of all demand sites are higher than required while minimizing the total number of vehicles. However, the probabilistic nature of the reliability constraint makes its modeling in mathematical terms very complex, which has been a significant

Published online: 05 July 2015

issue. Without considering stochastic characteristics, the model can give a solution that fails to guarantee reliability. Because the ALP has (1) a complex modeling structure that involves tough decision making and has an inherent stochastic nature, and (2) because the hypercube method is known to estimate reliability and analyze EMSs accurately, this study employs an iterative hypercube optimization framework for the purpose of suggesting the locations of ambulances with enhanced reliability. In the proposed iterative hypercube optimization protocol, (a) an optimization model is solved to determine the locations of ambulances; (b) the hypercube model is used to estimate the performance and parameters for the locations of ambulances obtained from the optimization procedures and to check whether the solution satisfies the reliability constraints; (c) if the solution is feasible from the hypercube outcome, the algorithm is terminated; (d) otherwise, the information obtained via the hypercube model is delivered to the optimization model; (e) the optimization model then finds a new solution based on the new information; and (f) the algorithm is repeated iteratively until it obtains a real, feasible solution. Additionally, we propose using the iterative simulation optimization algorithm for computational efficiency. It has the same framework as the iterative hypercube optimization algorithm except that the hypercube method is replaced by a discrete event-based simulation. The hypercube does not perform well when analyzing an EMS with a large number of ambulances due to its heavy computational burden, whereas this proposed simulation is less sensitive to the number of vehicles.

The rest of this paper is organized as follows: in section 2, we present a detailed survey of the relevant literature. In section 3, the study problem is defined using the hypercube approach. In sections 4 and 5, we describe the suggested iterative hypercube optimization framework and the approach using the simulation. The experimental results are presented in section 6, followed by a comprehensive conclusions section (section 7) and the cited references.

## 2 Literature review

Location problems in the existing literature can be categorized as the LSCP and the maximal covering location problem (MCLP). The LSCP finds the appropriate locations of ambulances to cover all of the demand sites within a specific distance while minimizing the number of ambulances (Toregas et al., [4]). The MCLP maximizes the coverage of ambulances for a preset number of available ambulances (Church and ReVelle, [5]). Brotcorne et al. [6] identified that ambulance locations decided according to the aforementioned models have specific shortcomings when the vehicles are busy. Daskin and Stern [7], Horgan and Revell [8], and Gendreau et al. [9] have suggested multiple coverage models to address this issue. It seems difficult to find the adequate number of ambulances to ensure a certain level of availability in these models (Shariat-Mahaymany et al. [10]).

Probabilistic models for the ALP address EMS requirements through considering the issue of ambulance unavailability; their approaches primarily use stochastic models and simulation models (to address the stochastic nature of the ALP) or use mathematical models (to address the decision-making process effectively).

Larson [11] proposed a hypercube model using a queuing framework for the ALP. The model accurately calculates ambulance utilization and the reliability level and is very helpful in determining overall performance of the EMS. In contrast, it is difficult to compute clear results for the total number of ambulances required. Larson [12], Jarvis [13], Goldberg and Szidarovszky [14], and Atkinson et al. [15, 16] have focused on the use of hypercube approximation algorithms methods to efficiently evaluate the EMS.

A number of studies have suggested the integrated hypercube model with optimization frameworks for the purpose of locating ambulances. There are two approaches in these studies. The first approach is to maximize several performance measures with the fixed number of vehicles. The problems to maximize the expected reliability (Batta et al., [17], Chiyoshi et al., [18], Saydam and Aytug, [19]), to maximize the number of nodes satisfying required reliability (Galvao et al., [20]), to minimize the expected traveling time (Geroliminis et al., [21, 22]), and to optimize various performance measures (Iannoni et al., [23]) with a fixed number of ambulances were investigated using the hypercube model incorporating solving algorithms such as meta-heuristics. The second approach is to minimize the number of vehicles satisfying a reliability requirement, which is usually called as PLSC P. Most of studies addressing PLSCP are based on a mathematical programming approach, and there are few studies that adopt hypercube approach. A variation of PLSCP, the dynamic available coverage location problem, was proposed by Rajagopalan et al. [24] using a combination of Jarvis' hypercube approximation and a tabu search algorithm. In their model, the reliability is estimated by hypercube approximation, however, it can be inaccurate in some cases.

Savas [25] used a simulation approach to study the EMS, and the average response time could be considerably improved by placing ambulances closer to high demand points and away from hospitals. Berlin and Liebman [26] conducted a detailed simulation analysis to analyze ambulance deployment in conjunction with a mathematical programming model. A model which predicts response time and pattern search was developed by Fitzsimmons [27]. The simulation framework developed by Borras and Pastor [28] measured the minimum local reliability levels.

In addition to the aforementioned hypercube models, several studies have been carried out to address the reliability concept by using mathematical approaches. Similar approaches have also used in these studies, namely, to maximize the expected reliability level with a fixed number of ambulances and to minimize the number of ambulances so that the reliability of each demand site is higher than required. For capturing reliability in these models, the utilization estimation of ambulances, the busy fraction, is sought as an integral aspect to take into account.

One of the first methods to approximate the utilization estimation of ambulances is the “given utilization method.” In this method, the utilization or the workload of an emergency vehicle is calculated before the number and locations of ambulances are determined, and the predetermined utilizations are incorporated into the models to capture the reliability concept. One of the first models addressing the MCLP in order to maximize reliability is the maximum expected covering location problem (MEXCLP) suggested by Daskin [29]. Repede and Bernardo [30] suggested an extension of MEXCLP, which covers the variation of travel time. Another extension of the MEXCLP that considers the stochastic travel time and the unequal vehicle utilizations is suggested by Goldberg et al. [31]. ReVelle and Hogan [2] suggested models to maximize the number of demand sites covered with a given reliability level $\alpha$ (called MALP I and MALP II). The PLSCP as suggested by ReVelle and Hogan [3] is the LSCP with a reliability level constraint. The utilization estimation of ambulances is more critical in the PLSCP than in the MCLP in their consideration of reliability, because inaccurate utilization estimation in the PLSCP may lead to an infeasible solution, and it can result in over- or underestimation of the objective value regarding the MCLP. ReVelle and Hogan [3] employed an area-specific busy fraction into their model, which shows the average utilization of the ambulances required to cover a demand site. The assumption that the utilization of all the ambulances covering a specific demand site will be the same is not a realistic assumption because all ambulances may have different utilization rates. Ball and Lin [32] suggested a model using a Poisson distribution that considers a site-specific busy fraction, that is, the utilization of ambulances within a specific site. Borras and Pastor [28] suggested two models: one using a queuing model as suggested by Marianov and Revelle [33] while employing a predetermined mean, and another one using a site-specific busy fraction (calculated by dividing all of the workload covered by a potential ambulance site by the available service time). The shortcoming of the given utilization method is that it considers the maximum possible demand while calculating the utilization of ambulances, which may lead to overestimation of the required number of ambulances.

The upper-bound utilization method suggested by Shariat-Mohaymany et al. [10] calculates the utilization of ambulances using a simplified assumption, that is, the reliability constraint is embodied in the model by limiting the utilization of ambulances to less than the maximum possible busy fraction. The advantage of this model is that it prevents overestimation of the required number of ambulances by calculating the utilization of ambulances without the maximum possible demand. Although this model considers utilization based on the locations and number of ambulances, it is not sufficient for finding a feasible solution because of two major deficiencies: (i) the identical ambulance workload assumption may not represent a real EMS, and (ii) the maximum-possible busy fraction is based on the independent operation of ambulances assumption; however, the operation of each ambulance obviously must have some dependence characteristic. These simplified assumptions may result in the violation of the reliability constraint. Lim et al. [34] suggested an iterative procedure of simulation optimization to handle the problem caused by the identical workload assumption. However, the interaction rule employed in their model is not capable of finding a feasible solution satisfying the required reliability in any instance.

To resolve this issue, this study suggests an iterative hypercube optimization algorithm that considers non-identical ambulance workloads and the maximum possible busy fraction based on an assumption of dependent ambulance operation for solving the PLSCP. The interaction rule utilized in this study is a novel framework in which the different interaction parameters, update rules, and termination conditions are exploited, which enables the suggested algorithm to find a feasible solution in any instance, and which is also efficient in terms of the number of ambulances. Also, we suggest an iterative simulation optimization algorithm to cover the heavy computational burden of the hypercube. It can solve the large-scale problem, which cannot be solved by the hypercube approach within a reasonable time, and its capability is almost equivalent to the iterative hypercube algorithm.

To summarize, the contributions of this paper are the following: 1) the proposed algorithms ensure finding feasible solutions of the PLSCP in any instance, whereas the other mathematical models and iterative algorithms cannot guarantee it in any instance, and 2) the computational experimental results show that the iterative simulation optimization algorithm is approximately identical to the one using the hypercube in terms of performance measures.

## 3 Problem modeling using hypercube

The objective function is to minimize the number of available ambulances required under a constraint, which is that the reliability levels of all nodes should be higher than the required level. There exists a maximum distance limit which the ambulance should be able to serve on an emergency call. The arrival times of emergency calls and service times have a stochastic characteristic; therefore, the model adapts the hypercube model suggested by Larson [11] shown in constraint (5–8), and the function of loss probability suggested by Iannoni et al. [23] shown in constraint (4). The following assumptions are employed in this paper to describe the emergency medical service system.

- Arrival times of emergency calls and service times follow exponential distributions
- An emergency call is assigned to an available vehicle at the shortest distance
- If there is no available vehicle within the maximum distance limit, a call is assigned to the private service system (the call is lost).
- Events, such as an arriving emergency call and finishing the services of an ambulance, are not allowed to occur simultaneously
- The average numbers of emergency calls per node vary independently
- The service times are identically distributed and do not depend on distance

Based on these assumptions, a hypercube model can be defined with a given number and locations of ambulances, which is determined by using an optimization model. Additionally, the optimization model seeks to find the optimal solution satisfying the required reliability constraints, where the reliability per node is computed by using the hypercube model. The notations and formulations for the PLSCP are as follows:

*Index:*

$i, j$ node index
$c$ ambulance index in each node

*Data:*

$\alpha$ required reliability level
$d_{ij}$ distance from node $i$ to node $j$
$S$ covering distance
$T$ average service time per call (hours)
$e_i$ average number of calls per hour in node $i$

*Decision Variables:*

$x_{ic}$ 1 if the $c^{th}$ ambulance is allocated in node $i$; otherwise, 0

*Hypercube notations:*

| | |
|---|---|
| $N=\Sigma_i\Sigma_c x_{ic}$ | the total number of ambulances |
| $k, l$ | vertices index of hypercube |
| $n$ | index for the allocated ambulances ($0 \le n \le N$) |
| $I_n$ | node index where ambulance $n$ is located (which can be defined by $x_{ic}$) |
| $O_{in}$ | preference of ambulance $n$ in node $i$ (for example, if $O_{in}=1$, then ambulance $n$ is the closest ambulance from node $i$, whereas, if $O_{in}=N$, the ambulance $n$ is the farthest one) |
| $CN_n = \{i \mid d_{iI_n} \le S\}$ | set of nodes in the covering distance from ambulance $n$ |
| $CA_i = \{n \mid d_{iI_n} \le S\}$ | set of ambulances in the covering distance from node $i$ |
| $B_k$ | vertices of hypercube ($0 \le k \le 2^{N\text{-}1}$), where each vertex is named as a sequential set of $N$ binary numbers taking the value of 1 when the vehicle is busy and of 0 when it is not |
| $b_{kn}$ | binary numbers in $B_k$; $b_{kn}=$ 1 if ambulance $n$ is busy in $B_k$; otherwise, $b_{kn}=0$ |
| $h_{kl}=\Sigma_n\lvert b_{ln}-b_{kn}\rvert$ | Hamming distance between two vertices $B_k$ and $B_l$ (for instance, the Hamming distance between vertices 0100 and 0110 is 1, while the distance between vertices 0110 and 0001 is 3) |
| $h^+_{kl}=\Sigma_n \max(b_{ln}-b_{kn}, 0)$ | upward Hamming distance from $B_k$ to $B_l$, which indicates the number of binary values switching from 0 to 1 |
| $h^-_{kl}=\Sigma_n \min(b_{ln}-b_{kn}, 0)$ | downward Hamming distance from $B_k$ to $B_l$, which indicates the number |

| | |
|---|---|
| | of binary values switching from 1 to 0 |
| $HN_{kl}$ | index of an ambulance that has different numbers between $B_k$ and $B_l$ (which is defined only if $h_{kl}$=1) |
| $FV_{kl} = \left\{ i \middle| \left( \Pi_{n\left(=o_{in}<o_{iHN_{kl}}\right)} b_{kn} \right) \times (1-b_{kHN_{kl}}) = 1 \& i \in CN_{HN_{kl}} \right\}$ | set of nodes where an arising emergency call will be assigned to ambulance $HN_{kl}$ (which is defined only if $h_{kl}$=1 and $h^+_{kl}$=1) |
| $q_{kl}$ | transition rate between two vertices $B_k$ and $B_l$ |
| $\pi\{B_k\}$ | steady-state probability of vertex $B_k$ |
| $lr_i$ | local reliability of node $i$ |

*Formulation:*

$$\text{minimize} \sum_i \sum_c x_{ic} \tag{1}$$

subject to

$$lr_i \geq \alpha \ \forall i \tag{2}$$

$$x_{ic} \in \{0,\ 1\} \forall i, c\cdot, \tag{3}$$

where $lr_i$ satisfies the following:

$$lr_i = \sum_{k \in \left\{ k \middle| \prod_{n \in CA_i} b_{kn} = 1 \right\}} \pi\{B_k\} \forall i \tag{4}$$

$$N = \sum_i \sum_c x_{ic} \tag{5}$$

$$\sum_k \pi\{B_k\} \cdot q_{kl} = 0 \qquad \forall l \tag{6}$$

$$\sum_k \pi\{B_k\} = 1 \tag{7}$$

$$q_{kl} \begin{cases} \sum_{i \in FV_{kl}} e_i & \forall\ k, l(h_{kl} = 1\ \& h^+_{kl} = 1\ ) \\ 1\ /\ T & \forall k, l(h_{kl} = 1\ \& h^-_{kl} = -1). \\ 0 & \forall\ k\ ,\ l\ \ (h_{kl} \geq 2) \\ -\sum_{l'(l' \neq k)} q_{kl'} & \forall\ k\ ,\ l\ (h_{kl} = 0) \end{cases} \tag{8}$$

The objective function represented in Eq. (1) is to minimize the total number of allocated ambulances. Constraint (2) ensures that the reliability per node is higher than the required reliability α. Constraint (3) requires that $x_{ic}$ takes a value of 1 or 0. Equations. (1)-(3) are involved with deciding the number and locations of vehicles, while Eqs. (4)-(8) are related to computing $lr_i$ representing the hypercube model. Formulation (4) computes $lr_i$ by summing the steady-state probabilities of the vertices where all the vehicles covering node $i$ are busy. The total number of ambulances $N$ is defined as $\sum_{ic} x_{ic}$ in formulation (5). Formulations (6) and (7) indicate the constraints to derive the steady-state probability $\pi\{B_k\}$. The computations of transition rate $q_{kl}$ are summarized in formulation (8). The upward transition rate $q_{kl}$ ($h_{kl}$=1 and $h^+_{kl}$ =1) is considered as the demand that makes the ambulance $HN_{kl}$ busy, which is computed as the sum of the arrival rates of the nodes of which the closest available ambulance is $HN_{kl}$. The downward transition rate $q_{kl}$ ($h_{kl}$=1 and $h^-_{kl}$=− 1) is the service rate $1/T$. There is no transition between vertices having Hamming distances ≥2. The transition rate of $q_{kk}$ is defined by -$\sum_{l'(l' \neq k)} q_{kl'}$.

The reliability level constraint in the PLSCP is complex and nonlinear, which makes it difficult to suggest its solving algorithm. One approach to handling this problem is to define the optimization and the hypercube model separately and develop an algorithm to find a solution via their interaction, which is described in section 4 of this paper.

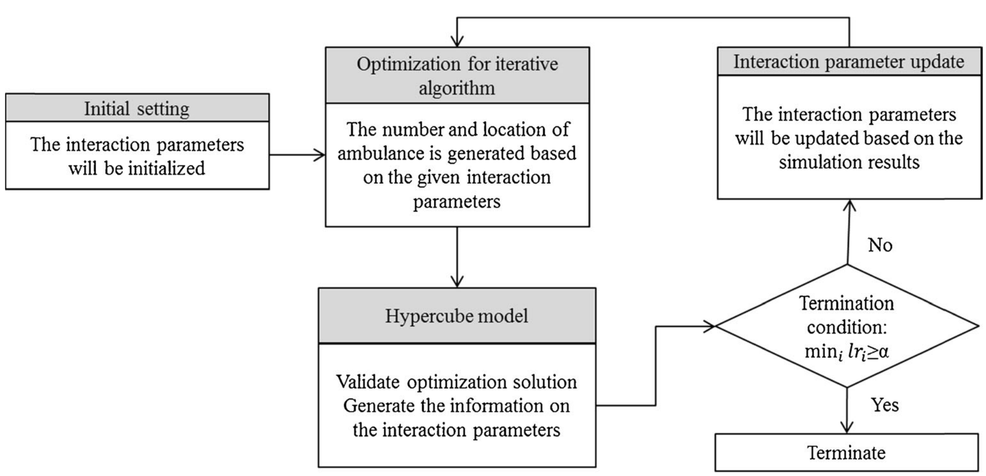


**Fig. 1** Procedure of Algorithm

## 4 Iterative hypercube optimization algorithm

The aim of this algorithm is to find a feasible solution in any instance that is based on the hypercube model results. In the PLSCP, a solution is feasible if and only if the reliability of each node resulting from the hypercube model is higher than required. This algorithm exploits the mathematical programming and hypercube in an iterative scheme to obtain a feasible solution because (1) the mathematical programming models for the ALP are able to find the number and locations of ambulances with consideration of a given constraint; however, it does not have the ability to accurately compute the reliability per node according to the locations of ambulances, and (2) the hypercube model is capable of precisely estimating the reliability per node with given locations of ambulances, but it does not have the ability to generate new locations of ambulances. Figure 1 shows the procedure of the suggested algorithm. The interaction parameters between the optimization and the hypercube are initialized in the initial step or setting. The optimization model solves the problem with the given parameters. The solution is validated via the hypercube, and the hypercube also gives information on the interacting parameters. The termination condition checks whether the feasibility requirement is satisfied. If the solution is found to be feasible as per the hypercube, the algorithm is terminated. Otherwise, it finds another solution after updating the interaction parameters.

### 4.1 Optimization model for the iterative algorithm

We developed the optimization for the suggested algorithm based on the upper-bound utilization method suggested by Shariat-Mohaymany et al. [10]. Their model is an efficient model regarding the number of ambulances; however, it is limited in its ability to obtain feasible solutions because of its assumptions of linearization. For obtaining a feasible solution, the following two facts should be considered. First, emergency calls arising from a demand site cannot be equally distributed to the ambulances covering the site. The percentage of ambulances per each node is dependent on the solution, so it cannot be determined before the solution is fixed. Second, the maximum possible busy fraction, which is the upper bound of utilization of ambulances to guarantee the reliability of each node that is to be higher than required, also depends on the locations of vehicles. This fact is based on the idea that, when an ambulance is busy, the probability that other vehicles close to it are busy increases because the demand covered by these ambulances is shared to the other ambulances. However, a complex nonlinear system will emerge to model the percent usage of ambulances per node and the maximum possible busy fractions in the optimization procedure. To handle these problems, we assume that the data about the ambulance's workload and the maximum possible busy fraction are generated by the hypercube and given to the optimization model. The suggested optimization model employs the following assumptions: (1) the required numbers of ambulances covering each node are the same; (2) all the maximum unavailability levels of ambulances remain the same (as in the work of Shariat-Mohaymany et al., [10]).

*Index:*

| | |
|---|---|
| $i, j$ | node index |
| $c$ | ambulance index in each node |
| $f$ | required number of ambulances per each node |
| $g$ | index for distance group ($1 \le g \le S/\theta$) |

*Data:*

| | |
|---|---|
| $\alpha$ | required reliability level |
| $d_{ij}$ | distance from node $i$ to node $j$ |
| $S$ | covering distance |
| $N_i=\{j \mid d_{ij} \le S\}$ | set of nodes in the covering distance from node $i$ |
| $T$ | average service time per call |
| $e_i$ | average number of calls per day in node $i$ |
| $\theta$ | distance interval (satisfying $S/\theta$=any integer) |
| $Pair_g=\{(i,j) \mid \theta \cdot (g-1) \le d_{ij} \le \theta\, g\}$ | subset of pair $(i, j)$ |
| $pre\_f$ | the pre-determined $f$ such that $r_f=1$ in the current iteration (in the first iteration, $pre\_f=0$) |

*Data from hypercube:*

| | |
|---|---|
| $mb_f$ | maximum possible busy fraction when the required number of ambulances per each node is $f$ (required to be observed via hypercube or simulation) |
| $z_{ij}^{hyper}$ | proportion of calls in node $i$ that are assigned to an ambulance in node $j$ (required to be observed via hypercube) |
| $min\_z_g = \min\limits_{(i,j)\in Pair_g} z_{ij}^{hyper}$ | the minimum of $z_{ij}^{hyper}$ in distance group $g$ |
| $max\_z_g = \max\limits_{(i,j)\in Pair_g} z_{ij}^{hyper}$ | the maximum of $z_{ij}^{hyper}$ in distance group $g$ |
| $sum\_z_g = \sum\limits_{(i,j)\in Pair_g} z_{ij}^{hyper}$ | the sum of $z_{ij}^{hyper}$ in distance group $g$. |

*Decision Variables:*

| | |
|---|---|
| $x_{ic}$ | 1 if the $c$th ambulance is allocated in node $I$; otherwise, 0 |
| $z_{ij}^{op}$ | proportion of calls in node $i$ that are assigned to an ambulance in node $j$ (to be determined) |

$r_f$ 1 if the required number of ambulance per each node is $f$; otherwise, 0

*Formulation:*

$$\text{minimize}\sum_i\sum_c x_{ic}$$

subject to

$$\sum_{j\in N_i}\sum_c x_{jc}\geq f\cdot r_f \quad \forall f,\ i \tag{9}$$

$$\sum_f r_f = 1 \tag{10}$$

$$\mathrm{M}(1-r_f) + mb_f\cdot\sum_c x_{jc}\geq\sum_i T\cdot e_i\cdot z_{ij}{}^{op} \quad \forall j,\ f \tag{11}$$

$$\sum_j z_{ij}{}^{op} = 1 \quad \forall i \tag{12}$$

$$\sum_{j\notin N_i} z_{ij}{}^{op}\leq 0 \quad \forall i \tag{13}$$

$$x_{ic}\leq x_{ic-1} \quad \forall i,\ c(c>1) \tag{14}$$

$$z_{ij}{}^{op}\geq \min\_z_g\cdot x_{j1} \quad \forall g, (i,\ j)\in P_g \tag{15}$$

$$z_{ij}{}^{op}\leq \max\_z_g\cdot x_{j1} \quad \forall g, (i,\ j)\in P_g \tag{16}$$

$$\sum_{(i,j)\in P_g} z_{ij}{}^{op}\leq sum\_z_g + \varepsilon \quad \forall g \quad (sum\_z_g\neq -1) \tag{17}$$

$$\sum_{(i,j)\in P_g} z_{ij}{}^{op}\geq sum\_z_g - \varepsilon \quad \forall g \quad (sum\_z_g\neq -1) \tag{18}$$

$$r_f = 1 \quad \forall f = pre\_f \tag{19}$$

$$x_{ic},\ r_f\in\{0,\ 1\},\ 0\leq z_{ij}{}^{op}\leq 1 \tag{20}$$

Constraint (9) is that the number of ambulances covering each node should be higher than required, and the required number of ambulances is determined via constraint (10). Constraint (11) indicates that the workload of an ambulance has to be lower than its maximum possible busy fraction. The maximum possible busy fraction depends on $r_f$, and that is why the term $M(1—r_f)$ is added in constraint (11) to make it work if and only if $r_f = 1$. Constraint (12) means that ever emergency call has to be assigned to some node. Constraint (13) ensures that emergency calls will not be assigned to nodes that lie outside the covering distance. Constraint (14) is that the $c^{\text{th}}$ ambulance could be allocated when the $(c–1)^{\text{th}}$ ambulance is allocated. Constraints (15)-(18) are the soft-bound constraints to make $z_{ij}{}^{op}$ similar to $z_{ij}{}^{hyper}$. In the presented model, $z_{ij}{}^{hyper}$ is given and updated based on the hypercube results. However, it is difficult to obtain the enough hypercube results to accurately estimate all of $z_{ij}{}^{hyper}$. Furthermore, $z_{ij}{}^{hyper}$ highly restricts $x_{ic}$, so the solution will be dominated by the solution in the previous iteration. These two facts motivated us to employ $g$, $\theta$ and $Pair_g$ for grouping $z_{ij}{}^{hyper}$ with distance. For example, there is a demand node, the high proportion of calls may be assigned to the nearest ambulance, which imply that $z_{ij}{}^{hyper}$ depends on the distance. Thus, pair $(i, j)$ is divided into $g$ groups according to the distance. For instance, pair $(i, j)$ satisfying $0\leq d_{ij}\leq\theta$ is included into group 1, and pair $(i, j)$ satisfying $\theta\leq d_{ij}\leq 2\theta$ is included into group 2. With the defined pair $(i, j)$, $z_{ij}{}^{hyper}$ is converted to $min\_z_g$, $max\_z_g$ and $sum\_z_g$, which does not fully represent each $z_{ij}{}^{hyper}$; however, they could act as soft boundaries on them. This approach could reduce the computational effort required to obtain a feasible solution, although it might result in the loss of optimality. Thus, if an ambulance is located in node $j$, at least the minimum and maximum values for $z_{ij}{}^{op}$ would be restricted by $min\_z_g$ and $max\_z_g$ according to its distance, as per constraints (15) and (16), respectively. Otherwise, $z_{ij}{}^{op}$ should be 0. Constraints (17) and (18) are boundary constraints on the sum of $z_{ij}{}^{op}$, which means the sum of $z_{ij}{}^{op}$ in distance group $g$ should be located in $[sum\_z_g—\varepsilon,\ sum\_z_g+\varepsilon]$, where $\varepsilon$ is the slack value. Furthermore, the fixed $r_f$ could reduce the solving time in the optimization procedure, and $r_f$ can be controlled by an uncomplicated rule without a decision process of optimization, after it is determined by the optimization model in the first iteration. Therefore, the parameter on $r_f$ is employed for computational efficiency and is denoted by $pre\_f$. With constraint (19), $r_f$ is predetermined unless $pre\_f = \varnothing$. Constraint (20) ensures that $x_{ic}$ and $r_f$ will be 0 or 1, and $z_{ij}{}^{op}$ will be floating between 0 and 1.

### 4.2 Initial setting

The interaction parameters are divided into three categories: $mb_f$, $z_{ij}{}^{hyper}$, and $r_f$. The parameters $mb_f$ and $z_{ij}{}^{hyper}$ are to be adjusted for the purpose of finding a feasible solution. The parameter $mb_f$ is given as $\sqrt[f]{1-\alpha}$ (as suggested by Shariat-Mohaymany et al., [10]). $min\_z_g$, $max\_z_g$, and $sum\_z_g$, which are the parameters for $z_{ij}{}^{hyper}$, respectively take the values of 0, 1, and $\varnothing$ in the initial setting, so the optimization model arbitrarily finds a solution without consideration of $z_{ij}{}^{hyper}$ in the first iteration. $pre\_f$ is set as $\varnothing$, as shown in Table 1.

**Table 1** Initialization of interaction parameters

| Category | Interaction parameter |
|---|---|
| $mb_f$ | $mb_f = \sqrt[f]{1-\alpha}$ |
| $z_{ij}{}^{hyper}$ | $min\_z_g$=0, $max\_z_g$=1, $sum\_z_g$=−1 |
| $r_f$ | $pre\_f$=0 |

### 4.3 Hypercube model for the iterative algorithm

The hypercube model framework is adopted for the purpose of validating the solution and updating the interaction parameters. The input variables of the hypercube model are the number and locations of ambulances, and its output variables are the interaction parameters, such as local reliability level and ${z_{ij}}^{hyper}$. To define ${z_{ij}}^{hyper}$, we adapt the concept of fraction of dispatch suggested by Geroliminis et al. [21]. The hypercube notations and Eqs. (4)-(8) defined in section 3 are also utilized for the hypercube in the iterative algorithm. Additionally, the interaction parameters not defined in section 3 are added as follows:

*Interaction parameters*

$$
{z_{ij}}^{hyper} = \sum\nolimits_{n(I_n=j \& j \in Ni)} \sum\nolimits_{k \in \left\{ k | \prod_{n'\left(o_{in'}<o_{in}\right)} b_{kn'}(1-b_{kn})=1 \right\}} \pi\{B_k\}/(1-lr_i)
$$

$MLR = \min_i lr_i$ : the minimum local reliability in total nodes

### 4.4 Termination condition

The aim of the suggested algorithm is to find a feasible solution, and the *MLR* is an important factor for the termination condition. The number of ambulances (which is a critical performance measure in the PLSCP) is also considered in the algorithm termination. The number of ambulances in the first iteration is assumed to be the ideal minimum number of ambulances suggested in the optimization model because $mb_f = \sqrt[f]{1-\alpha}$ is an ideal maximum value with the assumption of independent operation of ambulances. The procedure for the termination condition is defined as follows:

*Data:*

| | |
|---|---|
| *Iter* | index indicating current iteration number |
| *extra_iter* | extra iterations to find another feasible solution |
| *best_iter* | iteration index (when finding the best feasible solution) |
| *first_N* | number of ambulances in the first iteration |
| *best_N* | number of ambulances in the best solution up to the current iteration |

*Termination Condition:*

(1) $MLR \geq \alpha$ and $first\_N = \sum_i \sum_c x_{ic}$
(2) $iter \geq best\_iter + extra\_iter$

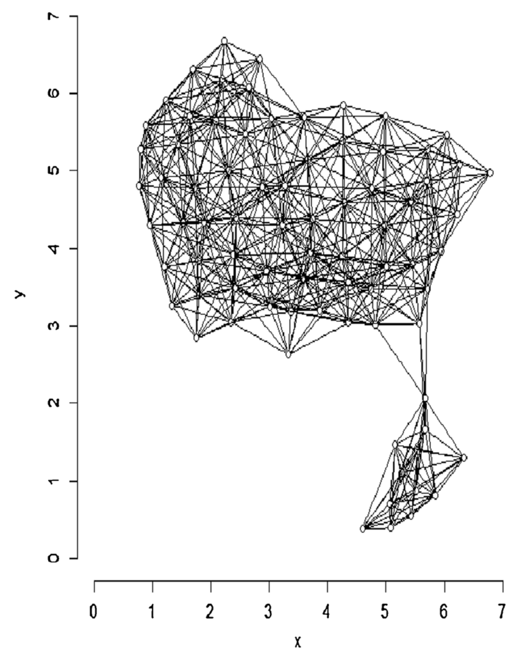


**Fig. 2** Graph of 79*79 network

The termination condition (1) is based on the assumption that the number of ambulances in the first iteration is an ideal minimum, as suggested in the optimization model. Thus, if a solution satisfying the termination condition (1) is found, there would be no need to search for another feasible solution and a lower number of ambulances, and so the algorithm would be terminated. The termination condition (2) is that the algorithm will be terminated after a particular number of searches after finding the best solution. If another best solution is found following the iterative procedure, the *best_N* and *best_iter* are then updated accordingly.

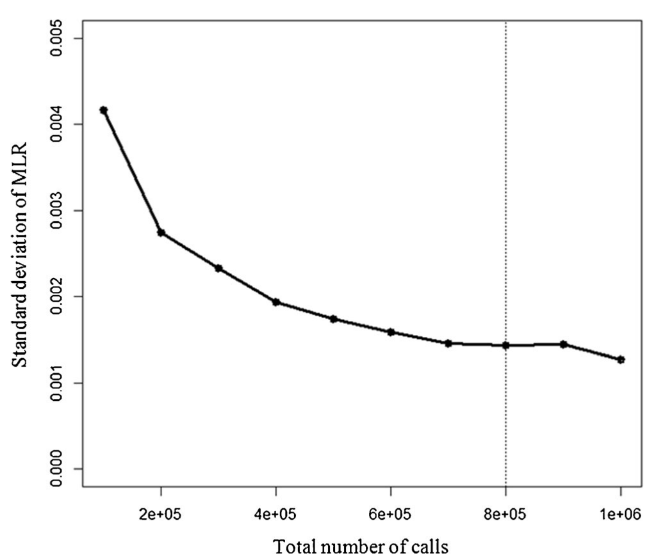


**Fig. 3** Standard deviation of MLR

**Table 2** Gap between simulation and hypercube models

| S | DS | α | *MLR* | | | Gap of $lr_i$ | | | Gap of $z_{ij}^{hyper}$ | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | MLR(S) | MLR(H) | Gap | Average | Std | Max | Average | Std | Max |
| 1.5 | 1 | .80 | 0.808 | 0.818 | 0.01050 | 0.00185 | 0.00214 | 0.01218 | 0.00216 | 0.00230 | 0.01126 |
| | | .825 | 0.826 | 0.826 | 0.00007 | 0.00137 | 0.00133 | 0.00557 | 0.00261 | 0.00259 | 0.01012 |
| | | .85 | 0.964 | 0.964 | 0.00018 | 0.00102 | 0.00091 | 0.00387 | 0.00258 | 0.00187 | 0.00883 |
| | | .875 | 0.963 | 0.965 | 0.00255 | 0.00096 | 0.00096 | 0.00521 | 0.00158 | 0.00183 | 0.00732 |
| | | .90 | 0.965 | 0.967 | 0.00188 | 0.00101 | 0.00098 | 0.00499 | 0.00224 | 0.00236 | 0.01194 |
| | | .925 | 0.961 | 0.963 | 0.00206 | 0.00093 | 0.00098 | 0.00567 | 0.00173 | 0.00206 | 0.01018 |
| | | .95 | 0.966 | 0.968 | 0.00177 | 0.00092 | 0.00092 | 0.00379 | 0.00193 | 0.00191 | 0.00842 |
| | | .975 | 0.977 | 0.976 | 0.00037 | 0.00060 | 0.00061 | 0.00319 | 0.00211 | 0.00198 | 0.00906 |
| | 2 | .80 | 0.923 | 0.925 | 0.00199 | 0.00158 | 0.00155 | 0.00644 | 0.00276 | 0.00229 | 0.01019 |
| | | .825 | 0.918 | 0.920 | 0.00252 | 0.00132 | 0.00118 | 0.00593 | 0.00289 | 0.00238 | 0.01053 |
| | | .85 | 0.897 | 0.903 | 0.00593 | 0.00141 | 0.00173 | 0.01266 | 0.00251 | 0.00256 | 0.01210 |
| | | .875 | 0.929 | 0.931 | 0.00210 | 0.00177 | 0.00193 | 0.01088 | 0.00280 | 0.00255 | 0.01315 |
| | | .90 | 0.924 | 0.927 | 0.00387 | 0.00152 | 0.00157 | 0.01087 | 0.00256 | 0.00254 | 0.01737 |
| | | .925 | 0.931 | 0.935 | 0.00356 | 0.00164 | 0.00128 | 0.00543 | 0.00280 | 0.00223 | 0.01122 |
| 3 | 1 | .80 | 0.826 | 0.833 | 0.00696 | 0.00271 | 0.00229 | 0.01117 | 0.00323 | 0.00272 | 0.01339 |
| | | .825 | 0.837 | 0.849 | 0.01203 | 0.00271 | 0.00233 | 0.01203 | 0.00348 | 0.00315 | 0.01453 |
| | | .85 | 0.859 | 0.873 | 0.01458 | 0.00239 | 0.00255 | 0.01458 | 0.00391 | 0.00386 | 0.02038 |
| | | .875 | 0.878 | 0.884 | 0.00597 | 0.00185 | 0.00158 | 0.00873 | 0.00381 | 0.00382 | 0.02113 |
| | | .90 | 0.956 | 0.963 | 0.00601 | 0.00132 | 0.00148 | 0.00894 | 0.00214 | 0.00215 | 0.01176 |
| | | .925 | 0.955 | 0.961 | 0.00538 | 0.00116 | 0.00111 | 0.00538 | 0.00256 | 0.00263 | 0.01426 |
| | | .95 | 0.955 | 0.960 | 0.00500 | 0.00135 | 0.00143 | 0.00753 | 0.00250 | 0.00250 | 0.02332 |
| | | .975 | 0.985 | 0.986 | 0.00186 | 0.00053 | 0.00062 | 0.00342 | 0.00212 | 0.00229 | 0.01271 |
| | | .99 | 0.991 | 0.993 | 0.00146 | 0.00056 | 0.00090 | 0.00734 | 0.00197 | 0.00234 | 0.01620 |
| | 2 | .80 | 0.898 | 0.903 | 0.00552 | 0.00215 | 0.00183 | 0.00931 | 0.00338 | 0.00326 | 0.01910 |
| | | .825 | 0.889 | 0.894 | 0.00507 | 0.00230 | 0.00235 | 0.01217 | 0.00293 | 0.00250 | 0.01568 |
| | | .85 | 0.891 | 0.897 | 0.00586 | 0.00184 | 0.00152 | 0.00586 | 0.00293 | 0.00259 | 0.01113 |
| | | .875 | 0.895 | 0.901 | 0.00629 | 0.00249 | 0.00295 | 0.02296 | 0.00332 | 0.00331 | 0.03236 |
| | | .90 | 0.905 | 0.910 | 0.00527 | 0.00146 | 0.00132 | 0.00673 | 0.00283 | 0.00264 | 0.01363 |
| | | .925 | 0.928 | 0.932 | 0.00430 | 0.00132 | 0.00144 | 0.00511 | 0.00243 | 0.00229 | 0.01255 |
| | | .95 | 0.958 | 0.958 | 0.00010 | 0.00118 | 0.00111 | 0.00457 | 0.00225 | 0.00224 | 0.01326 |
| | | .975 | 0.976 | 0.978 | 0.00161 | 0.00073 | 0.00066 | 0.00272 | 0.00216 | 0.00252 | 0.01927 |
| | | .99 | 0.993 | 0.994 | 0.00123 | 0.00040 | 0.00037 | 0.00142 | 0.00167 | 0.00203 | 0.01073 |
| | 3 | .80 | 0.832 | 0.843 | 0.01136 | 0.00316 | 0.00339 | 0.02712 | 0.00352 | 0.00451 | 0.03453 |
| | | .825 | 0.833 | 0.838 | 0.00546 | 0.00359 | 0.00419 | 0.02125 | 0.00359 | 0.00470 | 0.05734 |
| | | .85 | 0.872 | 0.872 | 0.00015 | 0.00250 | 0.00285 | 0.01546 | 0.00340 | 0.00609 | 0.08472 |
| | | .875 | 0.888 | 0.890 | 0.00263 | 0.00277 | 0.00323 | 0.02292 | 0.00332 | 0.00439 | 0.03474 |
| | | .90 | 0.913 | 0.918 | 0.00547 | 0.00231 | 0.00303 | 0.02177 | 0.00287 | 0.00494 | 0.05508 |
| | | .925 | 0.935 | 0.938 | 0.00233 | 0.00192 | 0.00383 | 0.03367 | 0.00305 | 0.00440 | 0.05416 |
| Average | | | | | 0.00424 | 0.00165 | 0.00175 | 0.01023 | 0.00270 | 0.00288 | 0.02020 |

### 4.5 Parameter update

The parameters $r_f$, $mb_f$, and $z_{ij}^{hyper}$ must be updated in the optimization model to obtain a better solution. The procedure for updating parameters is summarized in Appendix; it is composed of three main parts: updating parameters $r_f$, $mb_f$, and $z_{ij}^{hyper}$. The parameters of $r_f$ are $pre\_f$, $iter_f$, and $min\_amb_f$, which are initialized and updated in steps 1 and 2. $mb_f$ is updated in step 3, and step 4 updates according to the rules $min\_z_g$, $max\_z_g$, and $sum\_z_g$ (which are the parameters of $z_{ij}^{hyper}$). The update is performed with the results of the optimization ($r_f$, $x_{ic}$)

**Table 3** Comparison experiment between HYP-OPT and SIM-OPT

| S | DS | $\alpha$ | HYP-OPT | | | SIM-OPT | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | *MLR(H)* | *N* | *CT(*)* | *MLR(S)* | *MLR(H)* | *N* | *CT(*)* |
| 1.5 | 1 | .8 | 0.808 | 9 | 176(9) | 0.808 | 0.818 | 9 | 118(4) |
| | | .825 | 0.825 | 10 | 139(13) | 0.826 | 0.826 | 10 | 70(7) |
| | | .85 | 0.967 | 12 | 1941(12) | 0.964 | 0.964 | 12 | 43(9) |
| | | .875 | 0.965 | 12 | 293(1) | 0.963 | 0.965 | 12 | 5(1) |
| | | .90 | 0.967 | 12 | 300(1) | 0.965 | 0.967 | 12 | 4(1) |
| | | .925 | 0.963 | 12 | 295(1) | 0.961 | 0.963 | 12 | 4(1) |
| | | .95 | 0.968 | 12 | 229(1) | 0.966 | 0.968 | 12 | 5(1) |
| | | .975 | – | – | – | 0.977 | 0.976 | 14 | 2550(60) |
| | | .99 | – | – | – | 0.996 | – | 18 | 92(15) |
| | 2 | .80 | 0.925 | 12 | 75(1) | 0.923 | 0.925 | 12 | 4(1) |
| | | .825 | 0.920 | 12 | 75(1) | 0.918 | 0.920 | 12 | 4(1) |
| | | .85 | 0.903 | 12 | 78(1) | 0.897 | 0.903 | 12 | 5(1) |
| | | .875 | 0.931 | 12 | 77(1) | 0.929 | 0.931 | 12 | 5(1) |
| | | .90 | 0.927 | 12 | 76(1) | 0.924 | 0.927 | 12 | 5(1) |
| | | .925 | 0.930 | 12 | 699(8) | 0.931 | 0.935 | 12 | 25(6) |
| | | .95 | – | – | – | 0.957 | – | 15 | 219(37) |
| | | .975 | – | – | – | 0.988 | – | 18 | 67(13) |
| | | .99 | – | – | – | 0.990 | – | 18 | 14(3) |
| 3 | 1 | .80 | 0.828 | 4 | 7(1) | 0.826 | 0.833 | 4 | 6(1) |
| | | .825 | 0.836 | 4 | 5(1) | 0.837 | 0.849 | 4 | 11(2) |
| | | .85 | 0.859 | 5 | 62(11) | 0.859 | 0.873 | 5 | 79(10) |
| | | .875 | 0.877 | 5 | 35(2) | 0.878 | 0.884 | 5 | 170(21) |
| | | .90 | 0.964 | 6 | 79(10) | 0.956 | 0.963 | 6 | 116(10) |
| | | .925 | 0.959 | 6 | 3(1) | 0.955 | 0.961 | 6 | 12(2) |
| | | .95 | 0.957 | 6 | 4(1) | 0.955 | 0.960 | 6 | 9(1) |
| | | .975 | 0.981 | 8 | 685(82) | 0.985 | 0.986 | 8 | 383(54) |
| | | .99 | 0.991 | 8 | 10(1) | 0.991 | 0.993 | 8 | 7(1) |
| | 2 | .80 | 0.892 | 6 | 6(1) | 0.898 | 0.903 | 6 | 7(1) |
| | | .825 | 0.888 | 6 | 4(1) | 0.889 | 0.894 | 6 | 11(2) |
| | | .85 | 0.889 | 6 | 4(1) | 0.891 | 0.897 | 6 | 10(1) |
| | | .875 | 0.889 | 6 | 4(1) | 0.895 | 0.901 | 6 | 6(1) |
| | | .90 | 0.906 | 7 | 5(1) | 0.905 | 0.910 | 7 | 8(1) |
| | | .925 | 0.970 | 8 | 319(45) | 0.928 | 0.932 | 8 | 344(35) |
| | | .95 | 0.969 | 8 | 77(11) | 0.958 | 0.958 | 8 | 139(10) |
| | | .975 | 0.976 | 9 | 11(2) | 0.976 | 0.978 | 9 | 84(8) |
| | | .99 | 0.993 | 10 | 105(15) | 0.993 | 0.994 | 11 | 84(11) |
| | 3 | .80 | 0.810 | 11 | 104(5) | 0.832 | 0.843 | 11 | 21(2) |
| | | .825 | 0.825 | 11 | 347(11) | 0.833 | 0.838 | 11 | 61(5) |
| | | .85 | 0.866 | 12 | 288(3) | 0.872 | 0.872 | 12 | 8(1) |
| | | .875 | 0.886 | 12 | 2191(16) | 0.888 | 0.890 | 12 | 200(14) |
| | | .90 | – | – | – | 0.913 | 0.918 | 13 | 273(17) |
| | | .925 | – | – | – | 0.935 | 0.938 | 14 | 542(33) |
| | | .95 | – | – | – | 0.967 | – | 15 | 262(20) |
| | | .975 | – | – | – | 0.978 | – | 17 | 535(46) |
| | | .99 | – | – | – | 0.991 | – | 19 | 1302(112) |

*N* Number of ambulances

*CT* Computation time (sec)

*Number of iterations

Feasible solution is not found in 5 h or the case is out of memory

*MLR(H)* Minimum local reliability level from hypercube

*MLR(S)* Minimum local reliability level from simulation and the hypercube model ($z_{ij}^{hyper}$, $MLR$) in the current iteration.

## 5 Iterative simulation optimization algorithm

The iterative simulation optimization algorithm is the model having the same framework with the iterative hypercube optimization algorithm and adopting the simulation that is the substitution for the hypercube model. In this section, the simulation procedure of the suggested algorithm is only described, and the explanations on the other procedures are skipped because they are the same with ones in section 4. The input and output variables of the simulation are the same as those in the hypercube model because their purposes are identical.

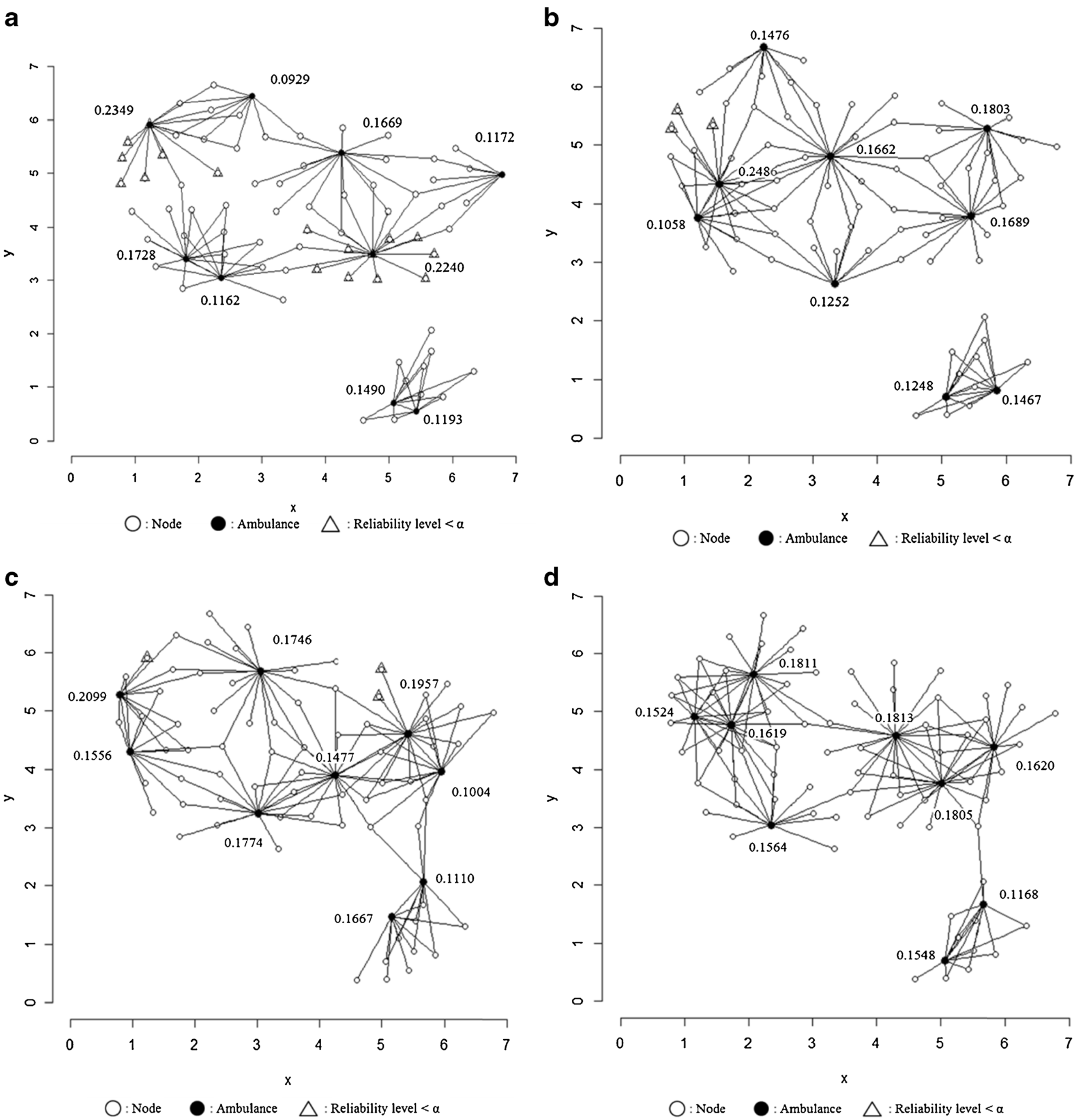


**Fig. 4** **a** First solution of SIM-OPT algorithm for DS=1, $S$=1.5, and $\alpha$=0.8. **b**. Second solution of SIM-OPT algorithm for DS=1, $S$=1.5, and $\alpha$=0.8. **c**. Third solution of SIM-OPT algorithm for DS=1, $S$=1.5, and $\alpha$=0.8. **d**. Fourth solution of SIM-OPT algorithm for DS=1, $S$=1.5, and $\alpha$=0.8


**Fig. 5** Iteration results of SIM-OPT algorithm (DS=4, $S$=3, and $\alpha$=0.925)

The simulation procedure is mainly composed of seven functions: initializing data, updating the call list, ordering the call list, assigning an ambulance, updating the next call time, the current time, and output data. At the start of the simulation, the running parameters and output data are initialized. In the next stage, the new emergency calls to be assigned are updated in the call list and sorted by the emergence time. After updating the call list, the emergency calls on the list are assigned to the closest available ambulances in the order of their emergence times. When a call is assigned to an ambulance, the available time of the ambulance is updated to the emergence time of the corresponding call plus its service time. When there is no available ambulance for a call, then the call will be lost. (It is assumed that the call is assigned to the private emergency system.) In the following step, the next call times of the nodes whose calls are on the call list are regenerated. After regenerating the next call times, it is required to check whether there are new emergency calls to be assigned. When there is no new emergency call, the current time is updated; otherwise, the procedures described above are repeated. The simulation is terminated when the total number of calls reached the predefined number of calls. In the last step, the output data are updated. The detailed procedure for the simulation is shown in Appendix.

## 6 Experimental results

In the first phase of this research, the simulation is validated. In the second phase, an iterative hypercube optimization model (HYP-OPT) and an iterative simulation optimization model (SIM-OPT) are compared. SIM-OPT and other mathematical programming approaches are compared to evaluate the performance of SIM-OPT in the last phase. The 79*79 network used by Serra [35], Borras and Pastor [28], Shariat-Mohaymany et al. [10], and Lim et al. [34] is employed in practice for comparison purposes. The experimental setup has 4 demand scenarios (DS), 2 covering distance scenarios ($S$=1.5 and 3 miles), 9 required reliability level scenarios ($\alpha$), and average service time $T$ defined as 0.75 (the total number of experiment cases thus is 72). In each demand scenario, the average numbers of calls for 79 nodes are between: [0.1365, 1.4425] for scenario 1 with average 0.7895, [0.2815, 2.3092] for scenario 2 with average 1.0000, [0.0147, 4.9852] for scenario 3 with average 2.7814, and [0.0264, 8.9309] for scenario 4 with average 4.8562. In these experiments, the performance measures of comparison are the number of ambulances, the *MLR*, and the feasibility, that is, the *MLR* is higher than required. Figure 2 shows the nodes and arcs having lengths less than 1.5 miles in the 79*79 network. The optimization model is coded using CPLEX 12.6 in JAVA to run, and the simulation model is also coded in JAVA. All experiments were performed on PC having Intel® Core 2 Quad 2.83 GHz processor with 8 GB RAM.

### 6.1 Simulation model validation

In the first step of the simulation model validation, the simulation period was set by analyzing the decrease in deviation of the MLR. In the second step, the validation of the simulation was carried out by determining the difference between the simulation and hypercube models in terms of the MLR, $lr_i$ and $z_{ij}^{hyper}$.

To obtain stable simulation results, the appropriate simulation period must be found. This study computed the standard deviations of MLR values, resulting in 20 repetitions of simulation runs while increasing the total number of calls by $10^5$. The locations of ambulances used for this experiment are the solutions obtained by SIM-OPT for each $\alpha$ scenario in the cases: DS=1, $S$=1.5. Figure 3 shows that the standard deviation of the MLR is converged with a total number of calls of $7*10^5$. Thus, the simulation period was set as the total number of calls of $8*10^5$.

**Table 4** The results of experiment $S$=1.5

| DS | α | UBUL | | | Lim et al. [34] | | | | SIM-OPT | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | *MLR(S)* | ≥α | *N* | *MLR(S)* | ≥α | *N* | *CT* | *MLR(S)* | ≥α | *N* | *CT(*)* |
| 1 | .80 | 0.812 | ○ | 9 | 0.815 | ○ | 9 | 505 | 0.808 | ○ | 9 | 118(4) |
| | .825 | 0.814 | X | 10 | 0.829 | ○ | 10 | 1643 | 0.826 | ○ | 10 | 70(7) |
| | .85 | 0.951 | ○ | 12 | 0.961 | ○ | 12 | 334 | 0.964 | ○ | 12 | 43(9) |
| | .875 | 0.950 | ○ | 12 | 0.962 | ○ | 12 | 50 | 0.963 | ○ | 12 | 5(1) |
| | .90 | 0.965 | ○ | 12 | 0.961 | ○ | 12 | 71 | 0.965 | ○ | 12 | 4(1) |
| | .925 | 0.954 | ○ | 12 | 0.961 | ○ | 12 | 47 | 0.961 | ○ | 12 | 4(1) |
| | .95 | 0.952 | ○ | 12 | 0.962 | ○ | 12 | 30 | 0.966 | ○ | 12 | 5(1) |
| | .975 | 0.966 | X | 12 | 0.975 | ○ | 13 | 2005 | 0.977 | ○ | 14 | 2550(60) |
| | .99 | 0.978 | X | 17 | 0.996 | ○ | 18 | 31 | 0.996 | ○ | 18 | 92(15) |
| 2 | .80 | 0.882 | ○ | 12 | 0.927 | ○ | 12 | 32 | 0.923 | ○ | 12 | 4(1) |
| | .825 | 0.909 | ○ | 12 | 0.920 | ○ | 12 | 28 | 0.918 | ○ | 12 | 4(1) |
| | .85 | 0.891 | ○ | 12 | 0.918 | ○ | 12 | 28 | 0.897 | ○ | 12 | 5(1) |
| | .875 | 0.894 | ○ | 12 | 0.921 | ○ | 12 | 31 | 0.929 | ○ | 12 | 5(1) |
| | .90 | 0.929 | ○ | 12 | 0.920 | ○ | 12 | 38 | 0.924 | ○ | 12 | 5(1) |
| | .925 | 0.903 | X | 12 | 0.924 | X | 12 | 127 | 0.931 | ○ | 12 | 25(6) |
| | .95 | 0.944 | X | 14 | 0.950 | ○ | 14 | 1159 | 0.957 | ○ | 15 | 219(37) |
| | .975 | 0.970 | X | 18 | 0.989 | ○ | 18 | 46 | 0.988 | ○ | 18 | 67(13) |
| | .99 | 0.986 | X | 18 | 0.987 | X | 18 | 60 | 0.990 | ○ | 18 | 14(3) |
| 3 | .80 | 0.798 | X | 17 | 0.873 | ○ | 18 | 695 | 0.812 | ○ | 17 | 75(10) |
| | .825 | 0.837 | ○ | 18 | 0.897 | ○ | 18 | 631 | 0.884 | ○ | 18 | 73(10) |
| | .85 | 0.864 | ○ | 18 | 0.882 | ○ | 18 | 38 | 0.869 | ○ | 18 | 7(1) |
| | .875 | 0.884 | ○ | 18 | 0.878 | ○ | 18 | 26 | 0.891 | ○ | 18 | 13(2) |
| | .90 | 0.891 | X | 19 | 0.905 | ○ | 20 | 589 | 0.902 | ○ | 19 | 174(33) |
| | .925 | 0.903 | X | 21 | 0.936 | ○ | 21 | 1707 | 0.937 | ○ | 21 | 184(32) |
| | .95 | 0.905 | X | 22 | 0.974 | ○ | 26 | 3163 | 0.951 | ○ | 23 | 152(23) |
| | .975 | 0.964 | X | 24 | 0.976 | ○ | 25 | 826 | 0.976 | ○ | 24 | 409(68) |
| | .99 | 0.978 | X | 26 | 0.993 | ○ | 30 | 1617 | 0.990 | ○ | 30 | 397(68) |
| 4 | .80 | 0.791 | X | 23 | 0.801 | ○ | 22 | 64 | 0.802 | ○ | 22 | 114(16) |
| | .825 | 0.791 | X | 23 | 0.882 | ○ | 24 | 749 | 0.826 | ○ | 23 | 79(10) |
| | .85 | 0.852 | ○ | 24 | 0.860 | ○ | 24 | 326 | 0.864 | ○ | 24 | 59(8) |
| | .875 | 0.851 | X | 24 | 0.878 | ○ | 24 | 34 | 0.886 | ○ | 24 | 115(17) |
| | .90 | 0.863 | X | 25 | 0.907 | ○ | 26 | 541 | 0.903 | ○ | 26 | 192(26) |
| | .925 | 0.897 | X | 28 | 0.930 | ○ | 27 | 1075 | 0.926 | ○ | 27 | 111(16) |
| | .95 | 0.908 | X | 30 | 0.951 | ○ | 30 | 3148 | 0.951 | ○ | 30 | 180(32) |
| | .975 | 0.933 | X | 34 | 0.986 | ○ | 37 | 1032 | 0.978 | ○ | 34 | 270(46) |
| | .99 | 0.966 | X | 35 | 0.991 | ○ | 38 | 4279 | 0.991 | ○ | 37 | 638(107) |

*N* Number of ambulances

*CT* Computation time (sec)

*Number of iterations

*MLR(S)* Minimum local reliability level from simulation

≥α: Feasibility of solution (○: Feasible, X: Infeasible)

The comparison experiments between the simulation and hypercube models were employed in this study for the purpose of simulation validation. The solutions from SIM-OPT having less than 14 vehicles were utilized for this comparison because the hypercube is not operated when the number of vehicles is more than 15, due to the memory limit of the computer. The models are compared in terms of MLR, $lr_i$, and $z_{ij}^{hyper}$.

The gap shown in the MLR column in Table 2 indicates the absolute difference between MLR(H) and

MLR(S). In all the cases, the gap is less than 0.01, and the average is 0.00424. The average, std, and max in the "gap of $lr_i$" and "gap of $z_{ij}^{hyper}$" columns represent the average, standard deviation, and maximum, respectively, of the absolute differences between the simulation and hypercube. The absolute differences of $lr_i$ are computed per node, and the absolute differences of $z_{ij}^{hyper}$ are calculated when $z_{ij}^{hyper}>0$. The average of the absolute differences of $lr_i$ is 0.00165 with a standard deviation of 0.00175, and the average of the maximum value is only 0.01023. Similarly, the average of the absolute differences of $z_{ij}^{hyper}$ is 0.00270 with a standard deviation of 0.00288, and the average of the maximum value is 0.0202. Thus, it is clear that the suggested simulation model provides similar results to those of the hypercube model.

### 6.2 Comparison experiments of HYP-OPT and SIM-OPT

The purpose of this comparison experiment is to show that HYP-OPT and SIM-OPT are approximately equivalent in terms of performance measures such as the number of ambulances and feasibility. The two algorithms are operated with the same initial settings for the interaction algorithm: *extra_iter*=5, $\theta$=0.15, $\lambda^-$=0.1, $\lambda^+$=1, and $\varepsilon$=2.

The comparison experiments of HYP-OPT and SIM-OPT are performed with a time limit of 5 h. As shown in Table 3, HYP-OPT found a feasible solution in the cases in which less than 12 vehicles were utilized. In the other cases, it failed to find a feasible solution because of reaching the time limit or the memory limit of the computer. The MLR(H) in the SIM-OPT column in Table 3 indicates the MLR values computed by the hypercube model for the solutions obtained by SIM-OPT. Due to the memory limit, the MLR(H) of SIM-OPT was calculated in the cases employing less than 14 ambulances. The conclusions derived from this comparison experiment are the following:

- HYP-OPT and SIM-OPT satisfied the reliability requirement in all cases using the same number of ambulances.
- The results of SIM-OPT show that MLR(S) and MLR(H) have similar values (which are examined in detail in section 6.3), and the solutions of SIM-OPT are also feasible in MLR(H).
- The solving time of HYP-OPT increased with the number of vehicles and iterations.
- The solving time of SIM-OPT depended on the number of iterations regardless of the number of vehicles.

After considering all the results of these experiments, it is apparent that SIM-OPT and HYP-OPT perform analogously.

Figures 4(a)–(d) show the locations of vehicles found when SIM-OPT solves the case: DS=1, $S$=1.5, $\alpha$=0.8. The circles filled with black in the corresponding figures represent ambulances, and they are connected to the nodes they cover. The nodes surrounded by triangle indicate that their reliabilities are lower than $\alpha$. The numbers around the circles filled with black indicate the utilization of the corresponding ambulance. The first location of ambulances searched by the optimization model with the initial settings is depicted in Fig. 4(a). This figure shows a significant difference among the utilizations of ambulances, indicating that greater demand was concentrated on specific vehicles. This result demonstrates that some vehicles (especially with utilization≥20 %) were so crowded that the nodes they covered did not satisfy the reliability requirement. After updating the interaction parameters based on the first results, the optimization model found the second solution, as shown in Fig. 4(b). Although the number of nodes that did not satisfy the required reliability was reduced, the reliabilities of three nodes covered by an ambulance whose utilization was 24.86 % were lower than required. Once again, the optimization found a third solution, as shown in Fig. 4(c), with the updated interaction parameters. The maximum utilization of ambulances

**Table 5** The efficiency of SIM-OPT in $S$=1.5

| Compared models | Feasibility | Number of cases | Average number of vehicles | | | > SIM-OPT | < SIM-OPT |
|---|---|---|---|---|---|---|---|
| | | | Comparing model | SIM-OPT | Gap | | |
| UBUL | O | 15 | 13.800 | 13.800 | 0.000 | 0 | 0 |
| | X | 21 | 21.524 | 22.000 | 0.476 | 2 | 7 |
| Lim et al. [34] | O | 34 | 19.059 | 18.794 | −0.265 | 7 | 2 |
| | X | 2 | 15.000 | 15.000 | 0.000 | 0 | 0 |

> SIM-OPT: the number of cases in which the number of allocated vehicles of SIM-OPT is less than in the compared model

< SIM-OPT: the number of cases in which the number of allocated vehicles of SIM-OPT is more than in the compared model

**Table 6** The results of experiment $S$=3.0

| DS | α | UBUL | | | Lim et al. [34] | | | | SIM-OPT | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | *MLR(S)* | ≥α | *N* | *MLR(S)* | ≥α | *N* | *CT* | *MLR(S)* | ≥α | *N* | *CT(*)* |
| 1 | .80 | 0.849 | ○ | 5 | 0.843 | ○ | 4 | 18 | 0.826 | ○ | 4 | 6(1) |
| | .825 | 0.859 | ○ | 5 | 0.842 | ○ | 4 | 22 | 0.837 | ○ | 4 | 11(2) |
| | .85 | 0.861 | ○ | 5 | 0.849 | X | 5 | 118 | 0.859 | ○ | 5 | 79(10) |
| | .875 | 0.886 | ○ | 5 | 0.877 | ○ | 5 | 66 | 0.878 | ○ | 5 | 170(21) |
| | .90 | 0.906 | ○ | 6 | 0.896 | X | 6 | 119 | 0.956 | ○ | 6 | 116(10) |
| | .925 | 0.925 | ○ | 7 | 0.948 | ○ | 6 | 317 | 0.955 | ○ | 6 | 12(2) |
| | .95 | 0.966 | ○ | 7 | 0.955 | ○ | 6 | 24 | 0.955 | ○ | 6 | 9(1) |
| | .975 | 0.964 | X | 7 | 0.964 | X | 7 | 99 | 0.985 | ○ | 8 | 383(54) |
| | .99 | 0.973 | X | 8 | 0.990 | ○ | 8 | 28 | 0.991 | ○ | 8 | 7(1) |
| 2 | .80 | 0.774 | X | 6 | 0.802 | ○ | 6 | 389 | 0.898 | ○ | 6 | 7(1) |
| | .825 | 0.835 | ○ | 7 | 0.878 | ○ | 6 | 316 | 0.889 | ○ | 6 | 11(2) |
| | .85 | 0.810 | X | 7 | 0.886 | ○ | 6 | 36 | 0.891 | ○ | 6 | 10(1) |
| | .875 | 0.908 | ○ | 7 | 0.885 | ○ | 6 | 22 | 0.895 | ○ | 6 | 6(1) |
| | .90 | 0.912 | ○ | 7 | 0.903 | ○ | 7 | 58 | 0.905 | ○ | 7 | 8(1) |
| | .925 | 0.913 | X | 7 | 0.934 | ○ | 8 | 1515 | 0.928 | ○ | 8 | 344(35) |
| | .95 | 0.928 | X | 8 | 0.940 | X | 8 | 1644 | 0.958 | ○ | 8 | 139(10) |
| | .975 | 0.982 | ○ | 9 | 0.977 | ○ | 9 | 174 | 0.976 | ○ | 9 | 84(8) |
| | .99 | 0.985 | X | 10 | 0.981 | X | 10 | 3272 | 0.993 | ○ | 11 | 84(11) |
| 3 | .80 | 0.854 | ○ | 12 | 0.804 | ○ | 11 | 506 | 0.832 | ○ | 11 | 21(2) |
| | .825 | 0.827 | ○ | 12 | 0.839 | ○ | 11 | 331 | 0.833 | ○ | 11 | 61(5) |
| | .85 | 0.819 | X | 12 | 0.851 | ○ | 12 | 57 | 0.872 | ○ | 12 | 8(1) |
| | .875 | 0.869 | X | 12 | 0.923 | ○ | 14 | 668 | 0.888 | ○ | 12 | 200(14) |
| | .90 | 0.911 | ○ | 13 | 0.925 | ○ | 13 | 378 | 0.913 | ○ | 13 | 273(17) |
| | .925 | 0.917 | X | 13 | 0.933 | ○ | 14 | 1859 | 0.935 | ○ | 14 | 542(33) |
| | .95 | 0.916 | X | 14 | 0.935 | X | 15 | 3536 | 0.967 | ○ | 15 | 262(20) |
| | .975 | 0.949 | X | 16 | 0.982 | ○ | 17 | 1885 | 0.978 | ○ | 17 | 535(46) |
| | .99 | 0.979 | X | 16 | 0.976 | X | 18 | 820 | 0.991 | ○ | 19 | 1302(112) |
| 4 | .80 | 0.809 | ○ | 17 | 0.840 | ○ | 17 | 586 | 0.804 | ○ | 16 | 26(3) |
| | .825 | 0.812 | X | 18 | 0.846 | ○ | 17 | 422 | 0.827 | ○ | 17 | 17(2) |
| | .85 | 0.857 | ○ | 18 | 0.853 | ○ | 17 | 324 | 0.853 | ○ | 17 | 114(11) |
| | .875 | 0.837 | X | 19 | 0.879 | ○ | 18 | 369 | 0.878 | ○ | 18 | 155(17) |
| | .90 | 0.832 | X | 19 | 0.916 | ○ | 19 | 542 | 0.903 | ○ | 19 | 173(17) |
| | .925 | 0.858 | X | 20 | 0.916 | X | 20 | 1059 | 0.926 | ○ | 20 | 520(25) |
| | .95 | 0.895 | X | 21 | 0.955 | ○ | 22 | 1080 | 0.951 | ○ | 22 | 284(23) |
| | .975 | 0.922 | X | 22 | 0.958 | X | 24 | 1927 | 0.975 | ○ | 24 | 740(54) |
| | .99 | 0.957 | X | 23 | 0.985 | X | 26 | 2248 | 0.991 | ○ | 27 | 1323(107) |

*N* Number of ambulances

*CT* Computation time (sec)

*Number of iterations

*MLR(S)* Minimum local reliability level from simulation

≥$\alpha$: Feasibility of solution (○: Feasible, X: Infeasible)

was reduced from 24.86 to 20.99 %; however, there remained nodes that did not satisfy the required reliability. In Fig. 4(d), the first feasible solution was found in the fourth iteration, showing that all the utilizations of vehicles were lower than 19 %.

Figure 5 shows the MLR and $mb_f$ depending on the iterations of SIM-OPT in the case: DS=4, $S$=3, and $\alpha$= 0.925. Looking at the corresponding graph, the optimization model found the location of 19 ambulances as the first solution, and its MLR was lower than $\alpha$. After

**Table 7** The efficiency of SIM-OPT in $S$=3.0

| Comparing model | Feasibility | Number of cases | Average number of vehicles | | | > SIM-OPT | < SIM-OPT |
|---|---|---|---|---|---|---|---|
| | | | Comparing model | SIM-OPT | Gap | | |
| UBUL | O | 16 | 8.875 | 8.250 | −0.625 | 10 | 0 |
| | X | 20 | 13.900 | 14.550 | 0.650 | 3 | 10 |
| Lim et al. [34] | O | 26 | 10.885 | 10.769 | −0.115 | 2 | 0 |
| | X | 10 | 13.900 | 14.300 | 0.400 | 0 | 4 |

> SIM-OPT: the number of cases in which the number of allocated vehicles of SIM-OPT is less than in the compared model

< SIM-OPT: the number of cases in which the number of allocated vehicles of SIM-OPT is more than in the compared model

finding the first solution, the SIM-OPT algorithm continuously found solutions with updated interaction parameters until the first feasible solution was found. Because the MLR values of the solutions found in the first 14 iterations were lower than required, the $mb_f$ gradually decreased. In the 15th iteration, the number of allocated vehicles increased by one, but its MLR was still lower than required. Finally, the first feasible solution was found in the 20th iteration, and the $mb_f$ increased as much as the MLR exceeded $\alpha$. The algorithm terminated when a better solution (a solution having a lower number of ambulances) was not found in an extra search.

### 6.3 Comparison experiments of SIM-OPT and mathematical programming approaches

The purpose of the second comparison experiment is to show that the suggested model is efficient with respect to the number of ambulances by comparing it with other mathematical programming models. The upper-bound unavailability location (UBUL) approach suggested by Shariat-Mohaymany et al. [10] and the iterative model suggested by Lim et al. [34], which have shown competent experimental results for the number of ambulances, were employed for this comparison. The MLR of each model was obtained via simulation, which provided similar results as with the hypercube described in section 6.1. The UBUL model is coded in CPLEX 12.6 to run. For the time efficiency of experiment, there are two minor modifications to UBUL in this experiment. Despite of the modification, the same UBUL model was solved in this experiment. At first, Shariat-Mohaymany et al. [10] solved the UBUL model several times with different $f$ settings to the same case for finding an appropriate $f$. In this paper, the decision variable $\boldsymbol{r_f}$ and the corresponding constraint (e.g., Constraint (9), (10) and the left side of Constraint (11)) are added for automatically finding the appropriate $f$. Secondly, the solutions of UBUL were obtained with the constraint that 'the total number of allocated ambulances $\geq \boldsymbol{N^{UBUL}}$', where $\boldsymbol{N^{UBUL}}$ is the optimal value shown in Shariat-Mohaymany et al. [10]. This experimental setting made us possible to obtain the optimal solution of UBUL in 8~10 s. The computational times for UBUL are not the time to solve the original UBUL, hence there is no computational time for the UBUL in Tables 4 and 6.

From the results of the experiments, detailed in Table 4, SIM-OPT shows a 100 % feasibility satisfaction rate compared to UBUL and Lim et al. [34], which show 46.9 and 94.4 % feasibility satisfaction rates, respectively. To evaluate the efficiency of SIM-OPT, the experimental cases are divided into those solutions of compared models that are feasible and those that are not. Table 5 shows the analysis of comparison of the experimental results provided in Table 4.

The first and second columns in Table 5 list the feasibilities of the compared models. For example, the second row in Table 5 (compared model=UBUL, feasibility=X) lists the results of cases in which UBUL finds infeasible solutions. The conclusions derived from Table 5 are the following:

- In the 15 cases in which UBUL found feasible solutions, SIM-OPT also found feasible solutions with the same numbers of ambulances
- In the 21 cases in which UBUL found infeasible solutions, SIM-OPT satisfied the required reliability in all the cases with an increase of only 0.4762 vehicles on average, and it found solutions that utilized more vehicles in 7 of 21 cases, and fewer vehicles in 2 cases.
- In the 34 cases in which Lim et al. [34] found feasible solutions, they employed 19.059 ambulances on average, and fewer vehicles were allocated compared to SIM-OPT in 2 cases. SIM-OPT used 18.794 ambulances on average, and the number of cases using fewer vehicles was seven.
- In the 2 cases in which Lim et al. [34] found infeasible solutions, SIM-OPT found feasible solutions with the same numbers of ambulances.

Table 6 lists the comparison experimental results when the covering distance was 3. The feasibility satisfaction ratios of UBUL, Lim et al. [34], and SIM-OPT were 44.44, 72.22, and 100 %, respectively (showing 2.25 and 1.38 times better performance). The results provided in Table 6 were analyzed according to the feasibilities of the compared model, as shown in Table 7.

Table 7 presents the efficiency of SIM-OPT when $S=3$. The findings from Table 7 are the following:

- In the 16 cases in which UBUL found feasible solutions, it employed 8.875 vehicles on average. In these cases, SIM-OPT used 8.25 vehicles on average, a decrease of 0.625. In 10 of the 16 cases, SIM-OPT satisfied the required reliability with fewer ambulances.
- In the 20 cases in which UBUL found infeasible solutions, the average number of vehicles used was 13.9. In these cases, SIM-OPT found feasible solutions using 14.55 vehicles on average, an increase of 0.65 compared to UBUL. Among these, SIM-OPT used more ambulances in 10 cases, and fewer in 3 cases.
- In the 26 cases in which Lim et al. [34] found feasible solutions, Lim et al. used 10.885 vehicles and SIM-OPT used 10.769 vehicles. In 2 of these cases, SIM-OPT performed better.
- In the 10 cases in which Lim et al. [34] found infeasible solutions, 13.9 ambulances were employed on average. For the same cases, SIM-OPT used 14.3 vehicles, an increase of 0.4. It found feasible solutions without increasing the number of ambulances in 6 of the 10 cases. In the other 4 cases, it found feasible solutions with one more vehicle.

The comparison experimental results support that SIM-OPT is as efficient as the other mathematical approaches, while it guarantees a feasible solution to the PLSCP for any case. When the other models found feasible solutions, SIM-OPT also found feasible solutions, and these used the same number or fewer ambulances, except for 2 cases with decreases of 0–0.625 vehicles on average. Otherwise, it satisfied feasibility with increases of 0–0.65 vehicles compared to the others. Comparing to Lim et al. [34], more information is delivered from simulation to optimization in the proposed SYM-OPT. In Lim et al. [34], only ${z_{ij}}^{hyper}$ is transferred from simulation to optimization, and the optimization model incorporates the information by minimizing the gap between ${z_{ij}}^{op}$ and ${z_{ij}}^{hyper}$ when an ambulance is allocated in node $i$. However, only interacting ${z_{ij}}^{hyper}$ between simulation and optimization cannot assure finding feasible solution. The suggested model uses $mb_f$ as interaction parameters as well as ${z_{ij}}^{hyper}$. Since $mb_f$ is closely related to MLR, an adjustment procedure to $mb_f$ is required for assuring MLR≥α. Furthermore, the information on ${z_{ij}}^{hyper}$ is incorporated in a different way from Lim et al. [34]. For instance, it uses the soft-bound constraint by grouping ${z_{ij}}^{hyper}$ according to the distance. Based on these differences, the suggested model can make the better results than the one by Lim et al. [34].

## 7 Conclusions

This paper suggested an iterative hypercube and simulation optimization algorithm for the purpose of finding the locations of ambulances that satisfy the reliability requirements. The optimization model of the suggested algorithms finds the number and locations of ambulances using simple linear constraints. The hypercube (simulation) model was used to validate the optimization model results, and a few parameters required to interact between the models for the purpose of validation were identified. These interaction parameters were updated iteratively until satisfaction of the termination conditions, thus ensuring a feasible solution. The comparison experiment of HYP-OPT and SIM-OPT showed that the two algorithms have approximately equivalent performances. The experimental results of SIM-OPT were compared with other mathematical programming algorithms, and it is evident from the statistics presented that the suggested algorithm is efficient in terms of the number of allocated vehicles. That is, it found a lower number of ambulances on average compared to the other algorithms for the cases in which the others found feasible solutions. In the other cases, it employed only 0.3815 more vehicles than the other models on average while satisfying the reliability requirement (computational time per case was observed to be<1 h). The limitations of this paper include the assumption that the required numbers of ambulances per demand node and the maximum utilizations of each ambulance are the same. The rules for updating the parameter ${z_{ij}}^{hyper}$ could be revised to impose a tight boundary on it in the optimization model. Future research is suggested to investigate an iterative algorithm that considers travel time and demand per period.

## 8 Appendix 1: pseudo-code of parameter update procedure

*Data:*

| | |
|---|---|
| *Iter* | index indicating the current iteration number |
| $iter_f$ | the number of iterations for which *pre_f*=*f* |
| $min\_amb_f$ | the minimum number of ambulances when $r_f=1$ |
| $\lambda^-$ | the step size for adjusting $mb_f$ when $MLR<\alpha$ |
| $\lambda^+$ | the step size for adjusting $mb_f$ when $MLR\geq\alpha$ |
| $pre\_min\_z_g$ | the $min\_z_g$ in the previous iteration |
| $pre\_max\_z_g$ | the $max\_z_g$ in the previous iteration |
| $pre\_sum\_z_g$ | the $sum\_z_g$ in the previous iteration |

*Parameter Update Procedure:*

Step 1 Initializing $iter_f$

If $iter = 1$

Set $iter_f$ to 0

Step 2 Updating the parameters of $r_f$

Step 2.1 Setting $pre\_f$

If $pre\_f = 0$

Set $pre\_f$ to $\sum_f f \cdot r_f$

Otherwise

Set $pre\_f$ to $pre\_f$

Step 2.2 Setting $min\_amb_f$,

If $iter_{pre\_f} = 0$

Operate optimization model with $r_{pre\_f+1} = 1$ and without constraints (15)–(19), and obtain the objective value $z'$

Set $min\_amb_{pre\_f+1}$ as $z'$ obtained in the previous step

Step 2.3 Changing $pre\_f$

If $\sum_i \sum_c x_{ic} \geq min\_amb_{pre\_f+1}$

Set $pre\_f$ to $pre\_f + 1$

Otherwise

Set $iter_{pre\_f}$ to $iter_{pre\_f} + 1$

Step 3 Updating the parameters of $mb_f$

If $MLR < \alpha$ and $iter_{pre\_f} \neq 0$

Set $mb_{pre\_f}$ to $mb_{pre\_f} - \lambda^{-} \cdot (\alpha - MLR))$

If $MLR \geq \alpha$ and $iter_{pre\_f} \neq 0$

Set $mb_{pre\_f}$ to $mb_{pre\_f} - \lambda^{+} \cdot (\alpha - MLR))$

Step 4 Updating the parameters of $z_{ij}^{hyper}$

If $iter_{pre\_f} = 0$

Set $min\_z_g$ to 0

Set $max\_z_g$ to 1

Set $sum\_z_g$ to –1

Otherwise

Step 4.1 Setting the parameters of $z_{ij}^{hyper}$ in the previous iteration

Set $pre\_min\_z_g$ to $min\_z_g$

Set $pre\_max\_z_g$ to $max\_z_g$

Set $pre\_sum\_z_g$ to $sum\_z_g$

Step 4.2 Setting the parameters of $z_{ij}^{hyper}$ in the current iteration

Set $min\_z_g$ to $\min_{(i,j) \in P_g} z_{ij}{}^{hyper}$

Set $max\_z_g$ to $\max_{(i,j) \in P_g} z_{ij}{}^{hyper}$

Set $sum\_z_g$ to $\sum_{(i,j) \in P_g} z_{ij}{}^{hyper}$

Step 4.3 Setting the parameters of $z_{ij}^{hyper}$

Set $min\_z_g$ to $(min\_z_g + (iter_{pre\_f} - 1) \cdot pre\_min\_z_g)/iter_{pre\_f}$

Set $max\_z_g$ to $(max\_z_g + (iter_{pre\_f} - 1) \cdot pre\_max\_z_g)/iter_{pre\_f}$

Set $sum\_z_g$ to $(sum\_z_g + (iter_{pre\_f} - 1) \cdot pre\_sum\_z_g)/iter_{pre\_f}$

*Parameter Update Procedure:*

The index $iter_f$, which indicates the number of iterations during $r_f$=1, is initialized in step 1. $iter_f$ is used to update $min\_amb_f$ and the interaction parameters $z_{ij}^{hyper}$. The parameters of $r_f$ will be updated in step 2. A predetermined $r_f$ can reduce the computational effort required for optimization, so this algorithm determines $r_f$ using a specific rule. The first $r_f$ is determined using an optimization model, and this $r_f$ remains fixed until the changing condition is satisfied, because the first $r_f$ has the capability of offering a solution with a minimum number of ambulances. If the increased number of ambulances reaches $min\_amb_{pre\_f+1}$, which is the minimum number of ambulances when $r_{pre\_f+1}$=1, the $pre\_f$ will be changed to $pre\_f$+1. The increased value to $pre\_f$+1 indicates that the number of ambulances covering each node has increased, which is better for the *MLR* empirically. Thus, the $pre\_f$ is increased when there is no loss in the number of ambulances. The procedures of computing $min\_amb_{pre\_f+1}$ and changing $pre\_f$ are presented in steps 2.2 and 2.3.

Updating $mb_f$ is presented in step 3 of the procedure. $mb_f$ has a close relationship with the *MLR*, which is summarized as follows: (1) a decrease in $mb_f$ tends to make the *MLR* higher and increase the number of ambulances, and (2) the increase in $mb_f$ tends to have the opposite effect. The rule for adjusting the $mb_f$ is that it is decreased by $\lambda^-$ ($\alpha$–*MLR*), if the *MLR* is lower than $\alpha$, where $\lambda^-$ is the step size. Otherwise, it is increased by $-\lambda^+$ ($\alpha$–*MLR*). How to update the interaction parameter $z_{ij}^{hyper}$ is shown in step 4. If $pre\_f$ is changed, the parameters of $z_{ij}^{hyper}$ will then be initialized. In other cases, the data for the parameters of $z_{ij}^{hyper}$ are accumulated by computing averages.

Using $mb_f$, the adjustment rule, this algorithm has the capability of finding a feasible solution. If the condition for the number of ambulances required to satisfy feasibility at any ambulance location is satisfied, such as with a large number *M*, and the *M* ambulances could be allocated, then this algorithm will definitely find a feasible solution because the algorithm decreases $mb_f$ until it finds a feasible solution, and an extremely decreased $mb_f$ causes it to find a solution with a number of ambulances larger than *M*.

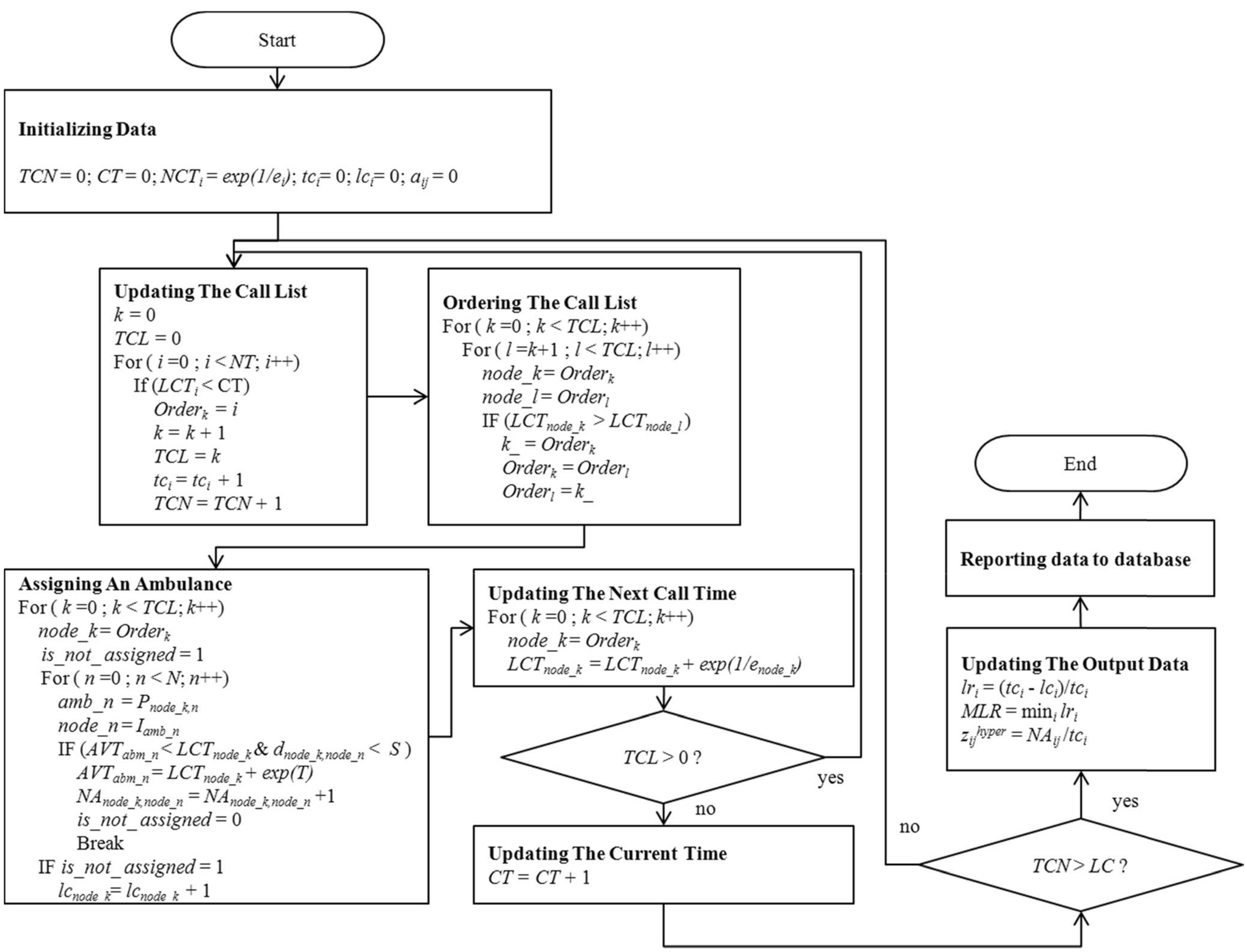


**Fig. 6** Framework of simulation

## 9 Appendix 2: framework of simulation

*Index*:

| | |
|---|---|
| *i, j* | indexes for nodes |
| *n* | index for allocated ambulances |
| *k, l* | indexes for calls |

*Given data*:

| | |
|---|---|
| $S$ | covering distance |
| $N$ | total number of ambulances |
| $NT$ | total number of nodes |
| $T$ | the mean of the exponential distribution for service time per call (min) |
| $e_i$ | the mean of the Poisson distribution for number of calls in a minute in node *i* |
| $d_{ij}$ | distance from node *i* to node *j* |
| $I_n$ | node index where ambulance *n* is located (which can be defined by $x_{ic}$) |
| $P_{in}$ | index of ambulance which is the $n^{th}$ closest ambulance from node *i* (which is defined by $x_{ic}$) |
| $LC$ | limit of call |

*Running parameters*:

| | |
|---|---|
| $TCN$ | total number of calls |
| $CT$ | current time (min) |
| $NCT_i$ | next call time of node *i* |
| $Order_k$ | index of node where the $k^{th}$ call arose in the current call list |
| $TCL$ | total number of calls in the call list |
| $AVT_n$ | available time of ambulance *n* |

*Output data*:

| | |
|---|---|
| $tc_i$ | the total number of calls arising in node *i* |
| $lc_i$ | the total number of lost calls in node *i* |
| $NA_{ij}$= 0 | the number of calls arising in node *i* and assigned to an ambulance in node *j* |
| $z_{ij}^{hyper}$ | the fraction of calls arising in node *i* and assigned to an ambulance in node *j* |
| $lr_i$ | the local reliability of node *i* |
| $MLR$ | the minimum local reliability in total nodes |

**Acknowledgments** This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. NRF-2014R1A2A2A03003874)

**Conflict of interest** The authors declare there are no conflicts of interest in this work.

**Human and Animal Rights** Human Participants and/or Animals are not involved in this work

## References

1. ReVelle C, Hogan K (1988) A reliability constrained siting model with local estimates of busy fractions. Environ Plan B 15(2):143–152. doi:10.1068/b150143
2. ReVelle C, Hogan K (1989) The maximum availability location problem. Transp Sci 23(3):192–200. doi:10.1287/trsc.23.3.192
3. ReVelle C, Hogan K (1989) The maximum reliability location problem and α-reliable p-center problem: derivatives of the probabilistic location set covering problem. Ann Oper Res 18(1):155–174. doi:10.1007/BF02097801
4. Toregas CR, Swain R, ReVelle CS, Bergman L (1971) The location of emergency service facilities. Oper Res 19(6):1363–1373. doi:10.1287/opre.19.6.1363
5. Church RL, ReVelle CS (1974) The maximal covering location problem. Pap Reg Sci Assoc 32(1):101–118. doi:10.1111/j.1435-5597.1974.tb00902.x
6. Brotcorne L, Laporte G, Semet F (2003) Ambulance location and relocation models. Eur J Oper Res 147(3):451–463. doi:10.1016/S0377-2217(02)00364-8
7. Daskin MS, Stern EH (1981) A hierarchical objective set covering model for emergency medical service vehicle deployment. Transp Sci 15(2):137–152. doi:10.1287/trsc.15.2.137
8. Hogan K, ReVelle CS (1986) Concepts and applications of backup coverage. Manag Sci 32(11):1434–1444. doi:10.1287/mnsc.32.11.1434
9. Gendreau M, Laporte G, Semet F (1997) Solving an ambulance location model by tabu search. Locat Sci 5(2):75–88. doi:10.1016/S0966-8349(97)00015-6
10. Shariat-Mohaymany A, Babaei M, Moadi S, Amiripour SM (2012) Linear upper-bound unavailability set covering models for locating ambulances: Application to Tehran rural roads. Eur J Oper Res 221(1):263–272. doi:10.1016/j.ejor.2012.03.015
11. Larson RC (1974) Hypercube queueing model for facility location and redistricting in urban emergency services. Comput Oper Res 1(1):67–95. doi:10.1016/0305-0548(74)90076-8
12. Larson RC (1975) Approximating the performance of urban emergency service systems. Oper Res 23(5):845–868. doi:10.1287/opre.23.5.845
13. Jarvis JP (1985) Approximating the equilibrium behavior of multi-server loss systems. Manag Sci 31(2):235–239. doi:10.1287/mnsc.31.2.235
14. Goldberg J, Szidarovszky F (1991) Methods for solving nonlinear equations used in evaluating emergency vehicle busy probabilities. Oper Res 39(6):903–916. doi:10.1287/opre.39.6.903
15. Atkinson JB, Kovalenko IN, Kuznetsov N, Mykhalevych KV (2006) Heuristic solution methods for a hypercube queueing model of the deployment of emergency systems. Cybern Syst Anal 42(3):379–391. doi:10.1007/s10559-006-0075-6
16. Atkinson JB, Kovalenko IN, Kuznetsov N, Mykhalevych KV (2008) A hypercube queuing loss model with customer-dependent service rates. Eur J Oper Res 191(1):223–239. doi:10.1016/j.ejor.2007.08.014
17. Batta R, Dolan JM, Krishnamurthy NN (1989) The maximal expected covering location problem: revisited. Transp Sci 23(4):277–287. doi:10.1287/trsc.23.4.277
18. Chiyoshi F, Galvao RD, Morabito R (2003) A note on solutions to the maximal expected covering location problem. Comput Oper Res 30(1):87–96. doi:10.1016/S0305-0548(01)00083-1
19. Saydam C, Aytug H (2003) Accurate estimation of expected coverage: revisited. Socio-Econ Plan Sci 37(1):69–80. doi:10.1016/S0038-0121(02)00004-6
20. Galvao RD, Chiyoshi F, Morabito R (2005) Towards unified formulations and extensions of two classical probabilistic location

models. Comput Oper Res 32(1):15–33. doi:10.1016/S0305-0548(03)00200-4
21. Geroliminis N, Karlaftis MG, Skabardonis A (2009) A spatial queuing model for the emergency vehicle districting and location problem. Transp Res Part B 43(7):798–811. doi:10.1016/j.trb.2009.01.006
22. Geroliminis N, Kepaptsoglou K, Karlaftis MG (2011) A hybrid hypercube–genetic algorithm approach for deploying many emergency response mobile units in an urban network. Eur J Oper Res 210(2):287–300. doi:10.1016/j.ejor.2010.08.031
23. Iannoni AP, Morabito R, Saydam C (2008) A hypercube queueing model embedded into a genetic algorithm for ambulance deployment on highways. Ann Oper Res 157(1):207–224. doi:10.1007/s10479-007-0195-z
24. Rajagopalan HK, Saydam C, Xiao J (2008) A multiperiod set covering location model for a dynamic redeployment of ambulances. Comput Oper Res 35(3):814–826. doi:10.1016/j.cor.2006.04.003
25. Savas E (1969) Simulation and cost-effectiveness analysis of New York's emergency ambulance service. Manag Sci 15(12):608–627. doi:10.1287/mnsc.15.12.B608
26. Berlin G, Liebman J (1974) Mathematical analysis of emergency ambulance locations. Socio-Econ Plan Sci 8(6):323–328. doi:10.1016/0038-0121(74)90036-6
27. Fitzsimmons JA (1973) A methodology for emergency ambulance deployment. Manag Sci 19(6):627–636. doi:10.1287/mnsc.19.6.627
28. Borras F, Pastor JT (2002) The ex-post evaluation of the minimum local reliability level: an enhanced probabilistic location set covering model. Ann Oper Res 111(1–4):51–74. doi:10.1287/mnsc.19.6.627
29. Daskin MS (1983) A maximum expected location model: Formulation, properties and heuristic solution. Transp Sci 17(1):48–70. doi:10.1287/trsc.17.1.48
30. Repede JF, Bernardo JJ (1994) Developing and validating a decision support system for locating emergency medical vehicles in Louisville, Kentucky. Eur J Oper Res 75(3):567–581. doi:10.1016/0377-2217(94)90297-6
31. Goldberg J, Dietrich R, Chen JM, Mitwasi MG (1990) Validating and applying a model for locating emergency medical services in Tucson, AZ. Eur J Oper Res 49(3):308–324. doi:10.1016/0377-2217(90)90402-W
32. Ball MO, Lin LF (1993) A reliability model applied to emergency service vehicle location. Oper Res 41(1):18–36. doi:10.1287/opre.41.1.18
33. Marianov V, ReVelle CS (1994) The queueing probabilistic location set covering problem and some extensions. Socio-Econ Plan Sci 28(3):167–178. doi:10.1016/0038-0121(94)90003-5
34. Lim YS, Kim SH, Lee YH (2012) A study of ambulance location problem applying the iterative procedure of simulation and optimization. J Korean Oper Res Manag Sci Soc 37(4):197–209. doi:10.7737/JKORMS.2012.37.4.197
35. Serra D (1989) The pq-Median Problem: Location and Districting of Hierarchical Facilities. Dissertation, John Hopkins University