



# A multiperiod set covering location model for dynamic redeployment of ambulances

Hari K. Rajagopalan<sup>a</sup>, Cem Saydam<sup>b,\*</sup>, Jing Xiao<sup>c</sup>

<sup>a</sup>*School of Business, Francis Marion University, Florence, SC 29501, USA*

<sup>b</sup>*Business Information Systems and Operations Management Department, The Belk College of Business, The University of North Carolina at Charlotte, Charlotte, NC 28223, USA*

<sup>c</sup>*Computer Science Department, College of Information Technology, The University of North Carolina at Charlotte, Charlotte, NC 28223, USA*

## Abstract

Emergency medical service (EMS) providers continually seek ways to improve system performance particularly the response time to incidents. The demand for ambulances fluctuate throughout the week, depending on the day of week, and even the time of day, therefore EMS operators can improve system performance by dynamic relocation/redeployment of ambulances in response to fluctuating demand patterns. The objective of the model is to determine the minimum number of ambulances and their locations for each time cluster in which significant changes in demand pattern occur while meeting coverage requirement with a predetermined reliability. The model is further enhanced by calculating ambulance specific busy probabilities and validated by a comprehensive simulation model. Computational results on experimental data sets and data from an EMS agency are provided.

© 2006 Elsevier Ltd. All rights reserved.

*Keywords:* Covering location problems; Hypercube model; Tabu search

## 1. Introduction

The goal of emergency medical services (EMS) is to reduce mortality, disability, and suffering in persons [1,2]. EMS administrators and managers often face the difficult task of locating a limited number of ambulances in a way that will yield the best service to a constituent population. Typically, calls originating from a population center are assumed to be covered if they can be reached within a time threshold. This notion of coverage has been widely accepted and is written into the EMS Act of 1973, which requires that in urban areas 95 percent of requests be reached in 10 minutes, and in rural areas, calls should be reached in 30 min or less [3].

The study of location models for ambulance location has undergone two different phases of evolution. During the first phase mostly deterministic models emerged while probabilistic and more realistic models emerged during the second. More recently the phenomenal advances in hardware, commercially available solvers (e.g., CPLEX [4]), and meta-heuristics have enabled researchers to develop increasingly more realistic and sophisticated models [2,5–10]. Earlier advances can be found in reviews by Schilling et al. [11], Owen and Daskin [12], and most recent and projected developments are in the latest review by Brotcorne et al. [1].

\* Corresponding author. Tel.: +1 704 687 2047; fax: +1 704 687 6330.

E-mail addresses: [hrajagop@unc.edu](mailto:hrajagop@unc.edu) (H.K. Rajagopalan), [saydam@unc.edu](mailto:saydam@unc.edu) (C. Saydam), [xiao@unc.edu](mailto:xiao@unc.edu) (J. Xiao).

Probabilistic location models recognize that any given ambulance may be busy when it is called. Such uncertainty can be modeled within the math programming formulation or using a queuing framework. Most models based on a mathematical programming approach tend to employ simplifying assumptions such as all units operate independently, yet have the same busy probability. For example, server independence and system-wide server busy probability is a common assumption used in Daskin's maximum expected coverage location problem (MEXCLP) [13] and ReVelle and Hogan's maximum availability location problem (MALP) [14]. However, not all models employ these simplifying assumptions. For example, in their second MALP formulation (MALPII), ReVelle and Hogan allow server busy probabilities to be different in various neighborhoods, sectors of a given region but not location or vehicle specific. Ball and Lin [3] take a similar approach found in MALP but instead of computing server busy probabilities in their reliability model, they compute the probability that a given demand point will not find an available server. The probabilistic location set covering problem (PLSCP) formulated by ReVelle and Hogan [15] minimized the number of servers needed to guarantee coverage for a city. This model like MALPII uses neighborhood server busy probabilities. Later Marianov and ReVelle [16] extended PLSCP using the assumption of neighborhood probabilities in MALPII to formulate queuing probabilistic location set covering problem (Q-PLSCP). They model each neighborhood as a multi-server loss system and calculate the neighborhood busy probabilities a priori and then use it as an input into the system.

True probabilistic models are based on spatially distributed queuing theory [17] or simulation [18] and they are by definition descriptive. Typically, they are employed to evaluate the vehicle busy probabilities and other performance metrics of a given allocation of ambulances. Larson's hypercube model [14,17,19] represents the most notable milestone for approaches using a queuing framework. The hypercube model and its various extensions have been found particularly useful in determining performance of EMS systems [2,17,20–27]. However, hypercube is computationally expensive. For  $m$  servers the number of simultaneous equations to solve would be  $2^m$ . For fleet sizes of about 10 or more this approach would be computationally impossible to solve with the present day computers. To solve this problem Larson developed an approximation to the hypercube problem [19] which would require solution only  $m$  simultaneous nonlinear equations for  $m$  servers. One of the assumptions used in Larson's approximation is that service times are exponentially distributed and identical for all servers, independent of the customers they are serving. Jarvis generalized Larson's approximation for loss systems (zero queue) by allowing service time distributions to be of a general type and may depend on both server and customer [28].

The literature of EMS models has been skewed towards the strategic decision making, where EMS managers make decisions on the location of ambulances or ambulance stations over a longer period of time. These models either focused on finding out the minimum number of ambulances needed to adequately cover an area or the maximum coverage that could be obtained given a set of ambulances. Both kinds of models were complimentary and serve different purposes. Minimization models were used on deciding the size of the ambulance fleet while maximization models were used to give an estimate of how good the system could perform over a period of time. Since a long term perspective was taken, the demand in these models was naturally assumed to be the same for all periods, therefore static. However, in reality, demand is not static but fluctuates throughout the week, day of week, and even hour by hour within a given day [2].

Dynamic redeployment models can aid managers make daily or even hourly plans to better respond to predictable demand fluctuations by time and space. There have been two earlier papers on relocation in the EMS literature [10,29]. Repede and Bernardo [29] extended MEXCLP for multiple time intervals to capture the temporal variations in demand and unit busy probabilities, hence, called their model TIMEXCLP. Their application of TIMEXCLP to Louisville, Kentucky resulted in an increase of coverage while the average response time decreased by 36%. The most recent and comprehensive dynamic relocation model was developed by Gendreau et al. [10]. The objective of their dynamic double standard model at time  $t$  (DDSM<sup>t</sup>) is to maximize backup coverage while minimizing relocation costs. There are several important considerations incorporated into this model. While the primary objective is to maximize the proportion of calls covered by at least two vehicles within a distance threshold, the model penalizes repeated relocation of the same vehicle, long round trips, and long trips. The model's input parameters are updated each time a call is received and DDSM<sup>t</sup> is solved. To solve this complex model, particularly at short time intervals, Gendreau et al. developed a fast tabu search heuristic implemented on eight parallel Sun Ultra workstations. Using real data from the Island of Montreal, their tests showed that the algorithm was able to generate new redeployment strategies for 95% of all cases. They randomly selected 33 problem instances and solved them using the tabu search. They then compared these solutions with those obtained using CPLEX and found that the worst case departure from optimality was only 2%. Apart from these studies, there has been very little work done concerning the periodic relocation (redeployment) of ambulances in an environment where demand and the location and number of available ambulances are changing.

Brotcorne et al. predict that current and future advances in this field are likely to be in probabilistic location models, dynamic redeployment models, and fast heuristics designed to solve generally large scale problem instances [1].

In this paper we formulate a multiperiod model for dynamic demand environments which minimizes the number of ambulances required while meeting predetermined ambulance availability requirements. We extend Marianov and ReVelle's Q-PLSCP [16] for multiple periods and incorporate ambulance specific busy probabilities using Jarvis' hypercube approximation algorithm [28] which further enhances the model's realism. We develop a tabu search heuristic to solve the model, validate the solutions by a comprehensive simulation model and provide computational statistics. We also apply the model to a city with significant demand variability both temporally and spatially. This article is organized as follows. Section 2 reviews PLSCP and Q-PLSCP, Section 3 presents the model and Section 4 details the search algorithm. Section 5 presents the numerical experiments and conclusions and suggestions for future work are discussed in Section 6.

## 2. Background

One of key developments in the literature is the PLSCP proposed by ReVelle and Hogan [15]. PLSCP minimizes the number of ambulances while guaranteeing a predetermined minimum coverage requirement with a certain reliability. Let  $x_i$  be the number of ambulances positioned at node  $i$ ,  $a_{ij} = 1$  if ambulance at node  $i$  is within time (or distance) threshold  $S$  of node  $j$ , 0 otherwise,  $p_k$  be the busy probability of all servers in area  $k$ ,  $\alpha$  be the reliability of guaranteed coverage, and  $b_k$  be the minimum number of servers required to cover each node. Using the chance constraints formulated by Charnes and Cooper [30] ReVelle and Hogan [14] show that the number of ambulances (servers) required can be computed by

$$b_k = \left\lceil \frac{\log(1 - \alpha)}{\log p_k} \right\rceil. \quad (1)$$

The PLSLCP is formulated as

$$\text{Minimize } \sum_{i=1}^n x_i \quad (2)$$

$$\text{Subject to } \sum_{i=1}^n a_{ij}x_i \geq b_j \quad \forall j \quad (3)$$

$$x_i \geq 0 \text{ and integer } \forall i. \quad (4)$$

Objective function (2) minimizes the number of servers to be deployed. Constraint (3) requires the number of servers covering a node  $j$  to be greater than or equal to  $b_j$  where  $b_j = b_k$ ,  $j \in k$ . Marianov and ReVelle [16] propose the Q-PLSCP as an extension to PLSCP. The key difference between Q-PLSCP and PLSCP is in how  $b_k$  is calculated using Eq. (1). In essence, PLSCP calculates an estimate of the  $p_k$  using the arrival rate and service rate in the area around the node  $i$  assuming independence among servers. In Q-PLSCP the  $p_k$ 's are still calculated for a neighborhood where each neighborhood is modeled as an M/M/s/0—loss queuing system. This allows Q-PLSCP to model server cooperation to a degree while avoiding more complex structures based on spatially distributed queuing theory [17,19,28].

## 3. The model

We formulate the dynamic available coverage location (DACL) model to determine the minimum number of ambulances and their locations for each time cluster in which significant changes in demand pattern occur while meeting coverage requirements with a predetermined reliability. An important feature of the DACL is the incorporation of the hypercube model [17] thus relaxing the simplifying assumptions that all servers have the same busy probability and operate independently. We follow an approach similar to Galvão et al. [31] but to minimize the computational effort we utilize Jarvis' hypercube approximation algorithm [28]. An added advantage of Jarvis' methodology is that it allows for server specific general service time distributions.

Let  $t$  be the index of time intervals from 1 to  $T$ ,  $x_{ik,t}$  be 1 if server  $i$  is located at node  $k$  at time  $t$  and  $m_t$  be the number of ambulances at time period  $t$ ,  $h_{j,t}$  be the fraction of demand at node  $j$  at time interval  $t$ ,  $n$  be the number of nodes in

the system, and  $c_t$  be the minimum expected coverage requirement at time  $t$ . Let  $p_{i,t}$  be the busy probability of a server at node  $i$  at time interval  $t$ ,  $\rho_t$  be the average system busy probability at time interval  $t$ ,  $P_0$  be the probability of having all servers free  $M/M/m/0$ -loss system,  $P_m$  be the probability of having all servers busy in an  $M/M/m$ -loss system, and  $Q(m, \rho_t, j)$  be the correction ( $Q$ ) factor for Jarvis' algorithm which adjusts the probabilities for server cooperation in the models. Let,

$$Q(m, \rho_t, j) = \frac{\sum_{k=j}^{m-1} (m-j-1)!(m-k)(m^k)(\rho_t^{k-j})P_0}{(k-j)!(1-P_m)^j m!(1-\rho_t(1-P_m))} \quad \forall j = 0, 1, \dots, m-1. \quad (5)$$

Also let,

$$y_{j,t} = \begin{cases} 1 & \text{if node } j \text{ is covered by at least one server with } \alpha_t \text{ reliability at time } t, \\ 0 & \text{if not,} \end{cases} \quad (6)$$

$$a_{ij,t} = \begin{cases} 1 & \text{if node } j \text{ is within the distance threshold of server at node } i \text{ during time interval } t, \\ 0 & \text{if not} \end{cases} \quad (7)$$

$$\text{Minimize } \sum_{t=1}^T \sum_{k=1}^n \sum_{i \in k} x_{ik,t} \quad (8)$$

$$\text{Subject to } \left[ \left[ 1 - \prod_{i=1}^{m_t} p_{i,t}^{\sum_{k=1}^n a_{ij,t} x_{ik,t}} Q \left( m_t, \rho_t, \sum_{j=1}^n \sum_{i=1}^{m_t} a_{jk,t} x_{ij,t} - 1 \right) \right] - \alpha_t \right] y_{j,t} \geq 0 \quad \forall j, t, \quad (9)$$

$$\sum_{j=1}^n h_{j,t} y_{j,t} \geq c_t \quad \forall t, \quad (10)$$

$$y_{j,t}, x_{ik,t} = \{0, 1\} \forall i, j, k, t. \quad (11)$$

Objective (8) minimizes the number of ambulances deployed. Constraint (9) tracks the nodes that are covered with the required (alpha) reliability. Constraint (10) ensures that total system wide coverage will be greater than  $c_t$  but in conjunction with the constraint (9) only the demand nodes that are covered with alpha reliability will be included in the system wide expected coverage statistic. For an in-depth analysis of availability and expected coverage metrics the reader is referred to Galvão et al. [5].

When system-wide or neighborhood busy probabilities are used in the models, these can be known or estimated a priori and thus the models may be solved using integer linear programming solvers and/or meta-heuristics with some computational effort. However, this model requires that we compute server busy probabilities at run times, therefore we developed a search algorithm to solve the problem. The main algorithm uses a tabu search [32] meta-heuristic to select the best sites for servers and Jarvis' hypercube approximation algorithm to compute server specific busy probabilities and the resulting coverage.

#### 4. A search algorithm

Location/relocation problems are typically NP-complete problems [25]. The size of the solution space of locating  $m$  response units in  $n$  zones is  $n^m$ . Because of the complex combinatorial nature of these problems, there have been various successful attempts to identify near-optimal solutions through the use of meta-heuristic search methods [33] such as; tabu search [10,34,35]; simulated annealing [31,36]; and evolutionary algorithms [8,9,37,38].

We developed a straightforward search algorithm to determine the minimum number of servers to meet the coverage and availability requirements. The algorithm starts with an estimated fleet size for the first time interval,  $m_1$ , calls a reactive tabu search (RTS) algorithm [39], a key heuristic which we describe below, to find the best set of locations for the given fleet size,  $m_t$ . In the RTS we utilize Jarvis' hypercube approximation algorithm to compute server busy probabilities and evaluate the resulting coverage using Eqs. (9) and (10). If the coverage exceeds the requirements then we decrement the fleet size otherwise increment the fleet size and run the algorithm. The best solution defined as the minimum fleet size that meets or exceeds the requirements is updated and stored as necessary.

For the next time interval, the algorithm takes number of servers and their locations found in the previous time interval as the initial locations for the RTS algorithm. This approach takes advantage of the nature of variations in demand for EMS calls. Real life EMS call demand patterns do not seem to have sudden and huge spatial shifts. Rather, the change is gradual and therefore using the best solution from time period  $t - 1$  as the initial solution for time period  $t$  actually improves the performance of search algorithm.

#### 4.1. A reactive tabu search algorithm

To determine the best locations for given fleet size we developed a variation of tabu search known as reactive tabu search [39] algorithm and tuned it to this problem domain. The major characteristic of a tabu search heuristic is its ability to escape from local optima by allowing a degradation of the objective. Hence there is a risk of cycling as previously visited solution may be revisited. To overcome this problem, a short term memory (tabu list) is used to prevent the algorithm from visiting recently visited solutions. Different tabu list definitions can be found in the literature. In RTS the tabu list is determined through feedback through the search. Similar to Gendreau et al. [35] the basic operation of our algorithm is moving an ambulance from node  $i$  to node  $j$ . All moves ( $i, j$  pairs) are stored in the long term memory. Initial tabu size is set to one. If a move in the long term memory occurs again, the tabu size increases to include that move and if the same move in the tabu list is not repeated for  $2 * m$  iterations, where  $m$  is the current number of servers, it is removed from the tabu list.

In our implementation we use a one dimensional data structure (an array) of size  $m_t * t + t$  where  $m_t$  is the number of response units (servers) and  $t$  is the number of time intervals. The array starts with the coverage value and the server locations for time interval 1 and then continues with the coverage value and location of the servers for all time intervals.

At iteration 0, we estimate the fleet size for each time interval,  $m_t$ , using system-wide average busy probability,  $\rho_t$ , average service rate,  $\mu_t$  and  $\lambda_t$ , the arrival rate, using  $m_t = \lambda_t / \rho_t \mu_t$ . The set of values for  $m_t$  gives the initial starting point of the algorithm. We then randomly generate server locations and run Jarvis' algorithm to compute server specific busy probabilities and the correction factor ( $Q$ ) to determine which demand nodes are covered with alpha reliability, Eq. (9). Next, we evaluate if the system wide coverage requirements are met using Eq. (10).

The basic operations in RTS or any other tabu search technique involves moving an ambulance from a node  $i$  to a node  $j$  where node  $j$  is the best location in the neighborhood. The pair ( $i, j$ ) become tabu as long as the tabu is effective. Neighborhood for this study is defined as the eight nodes surrounding the selected node. The first ambulance is selected for the basic operation and the best node from its neighborhood is selected only if the pair is not on the tabu list. We consider this as one iteration. We then systematically select the next ambulance and conduct a neighborhood search, and repeat this process until the last ambulance is selected after which the first ambulance is selected again. The size of the tabu list changes according to the exploration or exploitation pressure needed.

The stopping rule for this implementation of RTS is 100 iterations which we determined by running a set of sample problems for a long period of time (1000 iterations and more). Our experiments showed that the incremental gain after 100 iterations was minimal. The set of locations during the 100 iterations which resulted in the maximum available coverage is stored.

As mentioned above, if the best coverage after 100 iterations with a given number of servers ( $m_t$ ) is less than the required coverage, then we increase the number of servers and search for 100 more iterations and continue this process until the coverage constraint is met. If the coverage for the initial number of servers is more than the required coverage after tabu search then we reduce the number of servers by one and continue tabu search for 100 iterations until the coverage drops below the required coverage, in which case the most recent solution that meets the coverage constraint is the minimum fleet size.

#### 4.2. Look ahead procedure

During our initial experiments we discovered that despite the fact that Jarvis' hypercube approximation algorithm is efficient and fast, it still imposes a major computational overhead for this search process when we use it to calculate server specific busy probabilities for every relocation (move) while conducting the RTS. To minimize the computing time, we developed and tested a look-ahead procedure (LAP) within the RTS. Instead of using Jarvis' algorithm to compute server specific busy probabilities and the resulting coverage, the LAP uses a system wide average busy probability for 100 iterations which was determined after extensive testing on sample problems. At the end of 100

iterations, we run Jarvis' algorithm, calculate server specific busy probabilities,  $Q$  factor, and using Eq. (9) determine which nodes are covered by alpha reliability and with Eq. (10) we compute the resulting system wide expected coverage including only the nodes that meet the reliability requirement. However, we cannot assume that the current fleet size,  $m_t^*$ , and the resulting coverage are optimal, thus we run the search algorithm with the initial fleet size of  $m_t^*$  but without LAP, that is computing the ambulance specific busy probabilities every iteration via Jarvis' algorithm. Our results show that the RTS with LAP takes us very close to number of servers required to satisfy the coverage constraint generally 2–3 times faster than the algorithm without LAP. The details of the complete search algorithm are presented in the Appendix.

## 5. Computational results

We assume a hypothetical region (city) spanning 1024 square miles ( $32 \times 32$ ). We divide this region into 64- and 256-zones, representing medium and large scale problem sizes. We assume two time intervals, representing the hours before and during rush traffic. In  $t = 1$ , the calls are uniformly distributed across all zones whereas in  $t = 2$ , the call distribution reflects the shifts in call volume due to relocation of population from residential areas of the region to employment and perhaps education centers (Fig. 1). We randomly generate 10 problems each for 64-zone and 256-zone configurations, thus test our model on 20 problems. For coverage constraints the distance threshold is set to eight miles, and expected coverage requirement is set to 95% and the reliability is set to 90% for both time intervals. All tests are conducted on a 2.4 MHz. Pentium IV Dell PC with 512 Mb of RAM and the algorithms are coded in Java (jdk 1.4).

In our experimental design we use two metrics to compare the models and solution approaches: (a) quality of solution; and (b) run (CPU) time. Run time is self-explanatory. The quality of solution is measured by the coverage and the minimum number of servers required. We solved each of the test problems using the search algorithm with and without LAP. Fig. 2 displays the difference in the solution quality between the RTS and RTS-LAP for each problem which is on average 0.25% with maximum differences of 3.94% and  $-1.26\%$ , respectively.

Table 1 summarizes the run-time statistics for the two versions of the search algorithm. As anticipated, the search algorithm with the look ahead procedure solved the 64 zone problems on average in 0.30 min, which is 2.9 times faster than the search algorithm without the LAP. For 256 zone problems the algorithm with the LAP solved the problems on average in 2 min which is 2.1 times faster than the run times for the algorithm without LAP. The LAP procedure improves the run time for solution without any reduction in solution quality. A closer examination of the data shows that the significant gains in using LAP occur when the number of servers in the initial solution is very different from the number of servers in the final solution. If the number of servers in the initial solution is the same as the number of servers in the final solution, the algorithm with LAP procedure is actually slightly slower.

Next, we applied our model to Mecklenburg County, North Carolina, a region spread over approximately 540 square miles with a 2004 population of 801,137. The county's call demand distribution is known to fluctuate significantly

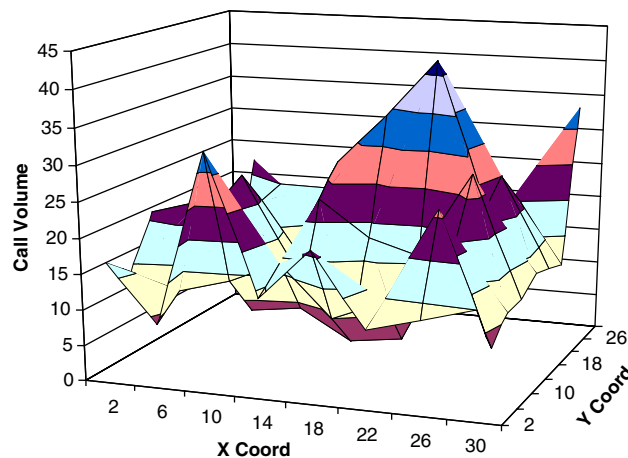


Fig. 1. Demand distribution for  $t = 2$ .

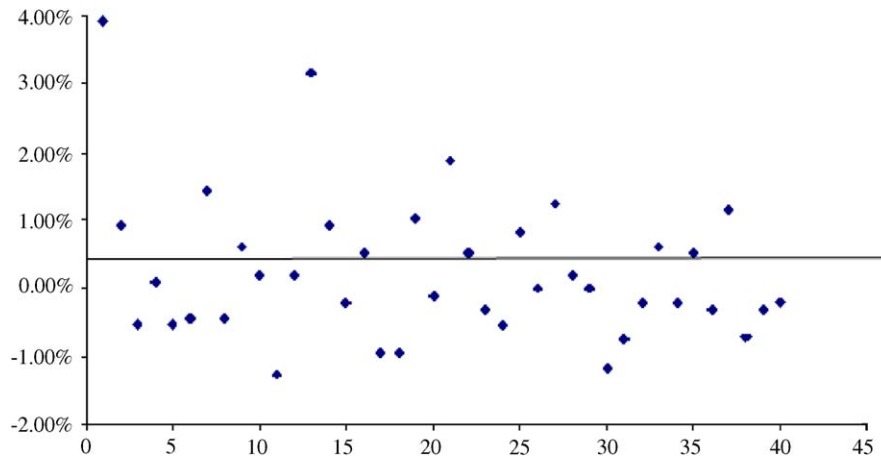


Fig. 2. Difference between the coverage from RTS and RTS-LAP.

Table 1  
CPU statistics in minutes (average, std. dev.)

Problem size	Search algorithm	
	RTS	RTS-LAP
64-zones	0.88, 0.63	0.30, 0.24
256-zones	4.20, 3.60	2.00, 0.98

Table 2  
Cumulative demand distribution per time interval for Mondays in Mecklenburg County

Interval no.	Time interval	Number of calls
1	0:00–3:00	635
2	3:00–6:00	467
3	6:00–9:00	990
4	9:00–12:00	1450
5	12:00–15:00	1513
6	15:00–18:00	1515
7	18:00–21:00	1259
8	21:00–24:00	913

by day of the week and time of the day which provides an opportunity for our research to be applied to real-life data thus increasing the chance of being implemented by interested parties and agencies. In 2004 the county received a total of 77,292 calls of which 61,630 of them were classified as emergency triggering ambulance dispatches. For this experiment we used data from one day of the week (Monday) for the entire year. There were 8,742 ambulance dispatches in the year 2004 for all Mondays distributed over the entire region. We chose eight, 3-h time intervals for our model. However, an EMS manager can choose, for example, 24 1-h blocks or even break the day into several time intervals with different durations. Essentially the user determines the appropriate duration of each time interval which could be based on experiential know-how or statistical analysis. Table 2 displays the yearly demand for each of the 3-h time intervals on Mondays.

We organized the county into 2 mile  $\times$  2 mile square nodes. This gave us a total of 168 demand nodes and we assumed that the ambulances can be posted in any one of the nodes except in the nodes that make up the boundary. Most of the boundary nodes are indeed not nearly the 2  $\times$  2 sq. miles but typically a third to at most half the full size of 4 sq. miles. This can be seen in Fig. 3 which displays the cumulative demand per node for all Mondays.



Table 3  
Predicted minimum fleet sizes and coverage statistics for Mondays

Time interval	No. of amb.	Predicted coverage	Simulated coverage (100 replications)			
			Average	std. dev.	Maximum	Minimum
1	17	97.01%	98.22%	0.48%	99.37%	97.01%
2	14	95.29%	97.92%	0.33%	98.72%	97.00%
3	20	95.56%	96.67%	0.31%	97.27%	96.06%
4	24	95.93%	97.35%	0.34%	98.21%	96.41%
5	23	95.84%	95.84%	0.38%	96.76%	94.91%
6	23	95.58%	96.53%	0.46%	97.36%	95.45%
7	22	95.08%	96.67%	0.36%	97.54%	95.63%
8	18	95.40%	95.76%	0.41%	96.50%	94.74%

used a weighted average of 58.58 min for service time, which is a service rate of 1.02/h. For coverage constraints the distance threshold was set to six miles, system-wide expected coverage requirement was 95% with a 90% availability requirement for nodes that are covered. We ran the ISA with the LAP to determine the fleet sizes and locations for the eight time intervals.

To measure the quality of a solution we developed a comprehensive discrete event driven simulation model also coded in Java. Given a fleet size and set of server locations, the simulation model provides an unbiased and accurate coverage statistic, thus serves as a benchmark. We fed the prescribed solutions (fleet size and server locations) to the simulation model to determine the expected coverage. This approach allows for measuring the accuracy of estimated (predicted) solutions which in turn reflects upon the model's ability to replicate the reality.

Using the locations prescribed by the model, then we ran the simulation using the actual call data from Mondays. We assumed that all calls come from the node centers and servers are located at the node centers. Given the large volume of data available (8742 observations), instead of using the empirical service time distribution, we randomly sampled from the actual service times recorded on Mondays in 2004. The simulation is replicated 100 times for each prescribed set of locations to give us an estimate of the resulting coverage. Table 3 below summarizes these results which compare the model predicted versus simulated "true" coverage statistics (average, standard deviation, maximum and minimum coverage). All eight instances of the problem resulted in higher than 95% coverage on average and are not very different from the predicted coverage. These results directly support the realism and accuracy of Larson's hypercube model [17,19] which we embed in our model using Jarvis' approximation algorithm [28]. These findings are consistent with the literature where locations generated by earlier models such as MEXCLP [9,20,40] were found to be quite robust but the expected coverage predictions could be off significantly.

The data from 2004 shows that to cover the peak periods on Mondays that the EMS agency deploys about 30 ambulances. Presently, using peak demand period data, they schedule two 12-h shifts and typically during the shift they do not to add or remove ambulances. Given our results, the agency is likely to consider alternative ways in scheduling daily shifts. As shown by our results the EMS agency can plan ahead with minimum fleet requirements in a given day (e.g., Monday) and utilize the surplus ambulances for non-emergency patient transport or as backups. More interesting possibility is to stagger the starting times for the fleet to approximately match or exceed the minimum fleet requirements prescribed by our model. That is, at the beginning of each time interval (minus the time to travel to their posts) varying numbers of ambulances can start their 12-h shifts. Furthermore, varying shift lengths such as 8-, 10-, and 12-h shifts could be devised to match the prescribed fleet sizes. These ideas are promising and challenging. Amiri's study of multi-hour service system design [41] applied to telecommunication systems provides some ideas that could be adapted in this domain. For future studies we suggest the development a decision support system which integrates the scheduling decisions with fleet sizes while meeting or exceeding the target coverage requirements.

## 6. Conclusions

We formulated a new model for dynamic environments with the objective of determining the minimum number of ambulances and their locations for each time cluster in which significant changes in demand pattern occur while meeting coverage availability requirements. Our model extends Marianov and ReVelle's Q-PLSCP [16] by incorpo-

rating ambulance specific busy probabilities similar to Galvão, Chiyoshi and Morabito [5]. To compute server specific busy probabilities we opted to utilize Jarvis' hypercube approximation algorithm which allows general service time distributions that are more common in real life data. We developed an incremental search algorithm to solve the model. The algorithm uses an implementation of reactive tabu search heuristic to determine the best set of locations for a given fleet size and uses Jarvis' hypercube approximation to compute server specific busy probabilities. To minimize the computational burden we devised a look ahead procedure which helped to solve the problems 2–3 times faster. We compared the quality of solutions via a simulation model. This approach helped us measure the realism of the solutions generated by the model using the ISA-LAP approach. The results showed that the model produces high quality solutions in reasonable computing times. We demonstrated how our model can be used in a real setting using data from a county with significant variations in calls for emergency services.

Given our promising results we suggest some future studies. Other meta-heuristics could very well be developed to solve the model [42]. We suggest a follow up study that compares the performance of several meta-heuristic implementations using computing times and solution quality. The model can be further enhanced by including penalties for successive redeployment of the same ambulances and avoiding long relocation trips introduced by Gendreau et al. [10]. With dynamic redeployment models how often and under what circumstances should the redeployment strategy needs to be updated remains an interesting research question. Since there are somewhat conflicting objectives, such as minimizing the fleet size while meeting a predefined hard coverage constraint, we can approach this problem from a multi-objective perspective and use the concept of Pareto domination criterion to analyze the compromise among these conflicting objectives and develop trade-off curves.

## Acknowledgments

The authors are indebted to the unknown reviewers for their critical review and suggestions which helped improve the paper.

## Appendix

Let,

$m_t$	the number of servers at time $t$
$T$	the number of time periods
$\lambda_t$	the arrival rate of calls at time $t$
$\mu_t$	the service rate at time $t$
$\rho_t$	the busy probability at time $t$
$V_t$	the search vector containing the locations of servers and the coverage at time $t$
$C_t(V_t)$	coverage of the search vector $V_t$
$In(V_t, x)$	initialize vector $V_t$ of size $x$ with randomly generated server locations
$RTS(V_t)$	reactive tabu search using vector $V_t$
$GetNeighbors(V_t, j)$	get the eight closest nodes to the node at $j$ th index in vector $V_t$
$N$	set of nodes selected by $Gen(V_t, j)$
$BestTabuCoverageWithJarvis(V_t(j), N)$	swap each node in set $N$ into vector $V_t$ index $j$ and return a vector which has the best coverage calculate using Jarvis algorithm embedded in Eqs. (16) and (17). Assuming node $x$ was swapped for node $y$ , the $(x,y)$ pair should be in the tabu list
$BestTabuCoverageWithoutJarvis(V_t(j), N)$	swap each node in set $N$ into vector $V_t$ index $j$ and return a vector which has the best coverage calculate using average busy probability in Eqs. (7) and (8). Assuming node $x$ was swapped for node $y$ , the $(x,y)$ pair should be in the tabu list

$$LAP = \begin{cases} True & \text{if LookAhead Procedure is used,} \\ False & \text{if not.} \end{cases}$$

*Main algorithm*

```

1.  Loop  $t \rightarrow 1$  to  $T$ 
2.     $m_t = \lambda_t / \rho_t \mu_t$ 
3.    If  $t \leftarrow 1$ 
4.       $V_t \leftarrow In(V_t, m_t + 1)$ 
5.    Else
6.       $V_t \leftarrow V_{t-1}$ 
7.    End If
8.     $V_t \leftarrow RTS(V_t, LAP)$ 
9.    If  $C_t(V_t) \geq c_t$ 
10.     Loop While  $C_t(V_t) \geq c_t$ 
11.        $m_t = m_t - 1$ 
12.        $V_t \leftarrow In(V_t, m_t + 1)$ 
13.        $V_t \leftarrow RTS(V_t, LAP)$ 
14.     End Loop
15.   Else
16.     Do Loop while  $C_t(V_t) \leq c_t$ 
17.        $m_t = m_t + 1$ 
18.       Go To Step 3
19.     End Loop
20.   End If
21.   If ( $LAP == True$ )
22.     Go to Step 3
23.   End If
24. End Loop
25. Return  $V_t$  for  $t = 1, \dots, T$ 

```

*Reactive tabu search algorithm  $RTS(V_t, LAP)$* 

```

1.   $Best \leftarrow CopyOf(V_t)$ 
2.  Loop  $i \leftarrow 1$  to 100
3.    Loop  $j \leftarrow 1$  to  $m_t$ 
4.       $N \leftarrow GetNeighbors(V_t, j)$ 
5.       $V_t^1 \leftarrow CopyOf(V_t)$ 
6.      If ( $LAP == False$ )
7.         $V_t^1(j) \leftarrow BestTabuCoverageWithJarvis(V_t(j), N)$ 
8.      Else
9.         $V_t^1(j) \leftarrow BestTabuCoverageWithoutJarvis(V_t(j), N)$ 
10.     End If
11.     If ( $\{V_t(j), V_t^1(j)\} \notin LongTermMemory$ )
12.        $LongTermMemory \leftarrow Add(\{V_t(j), V_t^1(j)\})$ 
13.        $Counter \leftarrow Counter + 1$ 
14.     Else
15.        $TabuList \leftarrow Add(\{V_t(j), V_t^1(j)\})$ 
16.        $Counter \leftarrow 0$ 
17.     End If
18.     If  $Counter == 2 * m_t$ 
19.        $Tabulist \leftarrow RemoveOldestElement(Tabulist)$ 
20.     End if
21.      $V(j) \leftarrow CopyOf(V_t^1(j))$ 
22.      $C_t(V_t) \leftarrow C_t(V_t^1)$ 

```

```

23.         If ( $C_t(Best) < C_t(V_t)$ )
24.              $Best \leftarrow CopyOf(V_t)$ 
25.         End If
26.     End Loop
27. End Loop
28. Return  $Best$ 

```

## References

- [1] Brotcorne L, Laporte G, Semet F. Ambulance location and relocation models. *European Journal of Operational Research* 2003;147:451–63.
- [2] Goldberg JB. Operations research models for the deployment of emergency services vehicles. *EMS Management Journal* 2004;1(1):20–39.
- [3] Ball MO, Lin LF. A reliability model applied to emergency service vehicle location. *Operations Research* 1993;41:18–36.
- [4] ILOG, ILOG Cplex 7.0 reference manual. 2000: ILOG.
- [5] Galvão RD, Chiyoshi FY, Morabito R. Towards unified formulations and extensions of two classical probabilistic location models. *Computers & Operations Research* 2005;32(1):15–33.
- [6] Karasakal O, Karasakal EK. A maximal covering location model in the presence of partial coverage. *Computers & Operations Research* 2004;31:1515–26.
- [7] Brotcorne L, Laporte G, Semet F. Fast heuristics for large scale covering location problems. *Computers & Operations Research* 2002;29: 651–65.
- [8] Aytug H, Saydam C. Solving large-scale maximum expected covering location problems by genetic algorithms: a comparative study. *European Journal of Operational Research* 2002;141:480–94.
- [9] Saydam C, Aytug H. Accurate estimation of expected coverage: revisited. *Socio-Economic Planning Sciences* 2003;37:69–80.
- [10] Gendreau M, Laporte G, Semet F. A dynamic model and parallel tabu search heuristic for real time ambulance relocation. *Parallel Computing* 2001;27:1641–53.
- [11] Schilling DA, Jayaraman V, Barkhi R. A review of covering problems in facility location. *Location Science* 1993;1(1):25–55.
- [12] Owen SH, Daskin MS. Strategic facility location: a review. *European Journal of Operational Research* 1998;111:423–47.
- [13] Daskin MS. A maximal expected covering location model: formulation, properties, and heuristic solution. *Transportation Science* 1983;17: 48–69.
- [14] ReVelle C, Hogan K. The maximum availability location problem. *Transportation Science* 1989;23:192–200.
- [15] ReVelle C, Hogan K. The maximum reliability location problem and alpha-reliable p-center problems: derivatives of the probabilistic location set covering problem. *Annals of Operations Research* 1989;18:155–74.
- [16] Marianov V, ReVelle C. The queuing probabilistic location set covering problem and some extensions. *Socio-Economic Planning Sciences* 1994;28:167–78.
- [17] Larson RC. A hypercube queuing model for facility location and redistricting in urban emergency services. *Computers & Operations Research* 1974;1:67–95.
- [18] Zaki AS, Cheng HK, Parker BR. A simulation model for the analysis and management of an emergency service system. *Socio-Economic Planning Sciences* 1997;31(2):173–89.
- [19] Larson RC. Approximating the performance of urban emergency service systems. *Operations Research* 1975;23:845–68.
- [20] Batta R, Dolan JM, Krishnamurthy NN. The maximal expected covering location problem: revisited. *Transportation Science* 1989;23:277–87.
- [21] Chan Y. *Location theory and decision analysis*. Cincinnati: South Western College Publishing; 2001.
- [22] Daskin MS. *Network and discrete location*. New York: Wiley; 1995.
- [23] Larson RC. *Urban operations research*. Englewood Cliffs, NJ: Prentice-Hall; 1981.
- [24] ReVelle C. Review, extension and prediction in emergency siting models. *European Journal of Operational Research* 1989;40:58–69.
- [25] Saydam C, Repede J, Burwell T. Accurate estimation of expected coverage: a comparative study. *Socio-Economic Planning Sciences* 1994;28(2):113–20.
- [26] Burwell T, Jarvis JP, McKnew MA. Modeling co-located servers and dispatch ties in the hypercube model. *Computers & Operations Research* 1993;20:113–9.
- [27] Takeda RA, Widmer JA, Morabito R. Analysis of ambulance decentralization in an urban medical emergency service using the hypercube queuing model. *Computers & Operations Research*, 2005, in press, corrected proof available online 17 May 2005. doi: 10.1016/j.cor.2005.03.022.
- [28] Jarvis JP. Approximating the equilibrium behavior of multi-server loss systems. *Management Science* 1985;31:235–9.
- [29] Repede J, Bernardo J. Developing and validating a decision support system for locating emergency medical vehicles in Louisville, Kentucky. *European Journal of Operational Research* 1994;75:567–81.
- [30] Charnes A, Cooper W. Chance constrained programming. *Management Science* 1959;6:73–9.
- [31] Galvão RD, Chiyoshi FY, Morabito R. Towards unified formulations and extensions of two classical probabilistic location models. *Computers & Operations Research* 2005;32(1):15–33.
- [32] Glover F, Laguna M. *Tabu search*. Boston, MA: Kluwer; 1997.
- [33] Osman IH, Laporte G. Metaheuristics: a bibliography. *Annals of Operations Research* 1996;63:513–628.
- [34] Gendreau M, Guertin F, Potvin JY, Taillard E. Parallel tabu search for real-time vehicle routing and dispatching. *Transportation Science* 1999;33:381–90.
- [35] Gendreau M, Laporte G, Semet F. Solving an ambulance location model by tabu search. *Location Science* 1997;5(2):75–88.

- [36] Chiyoshi FY, Galvao RD. A statistical analysis of simulated annealing applied to the p-median problem. *Annals of Operations Research* 2000;96:61–74.
- [37] Beasley JE, Chu PC. A genetic algorithm for the set covering problem. *European Journal of Operational Research* 1996;94:392–404.
- [38] Jaramillo J, Bhadury J, Batta R. On the use of genetic algorithms to solve location problems. *Computers & Operations Research* 2002;29:761–79.
- [39] Battiti R, Tecchiolli G. The reactive tabu search. *Journal on Computing* 1994;6(2).
- [40] Chiyoshi FY, Galvao RD, Morabito R. A note on solutions to the maximal expected covering location problem. *Computers & Operations Research* 2002;30:87–96.
- [41] Amiri A. The multi-hour service system design problem. *European Journal of Operational Research* 2001;128:625–38.
- [42] Rajagopalan HK, Vergara FE, Saydam C, Xiao J. Developing effective meta-heuristics for a probabilistic location model via experimental design. *European Journal of Operational Research*, 2005, in press, corrected proof available online 26 January 2006.