# An Efficient Heuristic for Real-Time Ambulance Redeployment

C.J. Jagtenberg, `jagtenbe@cwi.nl`
S. Bhulai, `s.bhulai@vu.nl`
R.D. van der Mei `mei@cwi.nl`

February 1, 2015

**Abstract**

We address the problem of dynamic ambulance repositioning, in which the goal is to minimize the expected fraction of late arrivals. The decisions on how to redeploy the vehicles have to be made in real time, and may take into account the status of all other vehicles and accidents. This is generally considered a difficult problem, especially in urban areas, and exact solution methods quickly become intractable when the number of vehicles grows. Therefore, there is a need for a scalable algorithm that performs well in practice.

We propose a polynomial-time heuristic that distinguishes itself by requiring neither assumptions on the region nor extensive state information. We evaluate its performance in a simulation model of emergency medical services (EMS) operations. We compare the performance of our repositioning method to so-called static solutions: a classical scenario in which an idle vehicle is always sent to its predefined base location. We show that the heuristic performs better than the optimal static solution for a tractable problem instance. Moreover, we perform a realistic urban case study in which we show that the performance of our heuristic is a 16.8% relative improvement on a benchmark static solution. The studied problem instances show that our algorithm fulfils the need for real-time, simple redeployment policies that significantly outperform static policies.

*keywords* Ambulances, Emergency medical services, Relocation, Redeployment.

2008 *MSC:* 90B99, 60K20, 90C90

# 1 Introduction

In a world where medical resources and budgets are limited, emergency medical services (EMS) managers are forced to rethink the way they spend both. Medical decisions aside, mathematical models can help them obtain more efficiency. They can also be helpful in understanding the effects of a certain decision (e.g., adding one extra vehicle, or changing the dispatch policy), which is otherwise difficult to oversee due to the stochastic nature of accidents. Typically, geographical aspects and service level agreements need to be taken into account when solving such problems.

In an EMS system, accidents occur randomly throughout the region[1]. Each accident needs to be served as soon as possible by an ambulance. The number of vehicles is typically limited, and vehicles are not always available due to serving other accidents. If an ambulance is not busy serving an accident, it is either on the road (driving), or stationed at one of the selected base locations. (Note that an idle ambulance can respond to an accident while still on the road, there is no need to return to a base location first.) Since minimizing the response time is critical in emergency situations, it is important to place ambulances in good positions with respect to the demand. This leads to the search for good base locations, as well as a good distribution of vehicles over the bases.

## 1.1 Related Work

In ambulance planning, models often use graph representations. Accidents can occur at the nodes, and there are certain distances (or driving times) between nodes. The travel times are assumed to be known in advance, and may be deterministic or stochastic (in which case they are only known in distribution). The goal is usually to maximize the fraction of accidents served within a certain (pre-determined) time. There are articles that search for the number of vehicles needed [17], the best base locations [4], and/or the best distribution of vehicles over the bases [5].

[1] Throughout this paper, we will use 'accidents' to refer to demand for ambulances. Accidents include medical incidents and are not limited to traffic collisions.

### Static Models

Mathematical models can be used at various stages of the EMS process. First, consider the *planning stage*. At this point, static models are often used to describe the problem. Here ‘static’ means that each ambulance is sent to its own home base whenever it becomes idle. These models can be used to determine the optimal locations of bases, as well as the number of vehicles needed per base.

Early research in ambulance planning focused on deterministic location problems [4], [8]. These formulations ignore the stochastic aspects of an EMS system, typically by assuming that one vehicle, or a constant number of vehicles, is always sufficient to cover the demand points. Later, research turned to probabilistic static models. A well-known example is the maximum expected covering location problem formulation (MEXCLP) [5]. In this formulation there is a limited number of vehicles that need to be distributed over a set of possible base locations. Each vehicle is modeled to be unavailable with a pre-determined probability. For a more detailed description of this model, we refer the reader to Section 2.

Over the years, several variants of MEXCLP have been published by different authors [7], [15]. These models are generally considered to give good static solutions. (Note a static solution can be defined by giving the location of the ‘home base’ for each vehicle.) The downside of static policies is that they do not utilize all possibilities, e.g., real-time information, to obtain good coverage. Clearly, the assumption of a vehicle belonging to a specific base is unnecessary in real life. Using the models above, one can attempt to find the optimal policy within the solution space of static policies. However, in the space of all policies, this will almost always be suboptimal.

### Dynamic Models

Dynamic models are used to find good (re)distributions of vehicles when a number of ambulances is busy responding to accidents. I.e., they look for repositioning strategies, which stand in contrast to strategies where every ambulance is sent back to its ‘home base’ after serving an accident. The first of such models can be found in [6], using tabu search. This shifted focus of research was accompanied with an increasing number of EMS systems using a dynamic allocation of vehicles to bases. Surveys of North American EMS operators showed that the percentage of operators who used a dynamic strategy increased from 23% in 2001 [3] to 37% in 2009 [19] (see also [1]). This indicates that the EMS community is becoming more aware that a dynamic policy can help them achieve greater service without increasing capacity.

Dynamic models usually do not search for good base locations, but instead consider the bases as a given, fixed set. The redeployment policies that have been published so far are roughly dividable in two subclasses, which we will (very generalizing) refer to as lookup tables and real-time optimization.

*Lookup tables.* The models in this class are typically looking for an optimal configuration for each number of available ambulances. A recent example can be found in [1]. The job of steering the set of available vehicles towards this configuration is usually left to the dispatchers. Unfortunately, poorly executed redeployment can devaluate even the most crafty policy. Even if the decision of how to move the vehicles in order to obtain the required configuration is part of the mathematical solution, this approach altogether requires a lot of ambulance movements. This increases the work load on the ambulance crew, which is a downside in many realistic situations. Furthermore, note that in busy regions, where the number of idle ambulances changes rapidly, the system will not be in compliance with the lookup table for most of the time.

*Real-time optimization.* On the other hand, there are various papers that model the randomness in the system explicitly, for example, by formulating the problem as a Markov decision process. When the model has only a few ambulances, one can solve it using exact dynamic programming [21].

When the state space grows, for example due to the number of vehicles considered, the problem quickly becomes intractable. In those cases we need to turn to alternative solution methods. Successful approaches include approximate dynamic programming [13]. Here, the state space is modelled rather elaborately, and the authors need advanced mathematical methods to solve the problem. Furthermore, it requires a mechanism to tune parameters to the use case, which is time consuming to both implement and execute. For one large city, the tuning process can take as long as one year. It remains possible to calculate the repositioning decision in real time, because these heavy computations are done in a preparatory phase. Furthermore, the authors try to speed up the tuning process, for example by using the so called post decision state. (For an elaborate discussion of the post decision state, see [14].) For the use case of the city of Melbourne described in [12], this reduced the computation time from approximately one year to 12 hours. Although this demonstrates the power of the post decision state, the remaining 12 hours should also highlight the complexity of the method. The heavy pre-computations and the need for an expert to implement this, make this method inaccessible and impractical.

Furthermore, the performance of the approximate dynamic programming approach is highly dependent on the choice of base functions. The base functions as defined in [12] are elegant, but unlikely to work well in general. That is because the underlying idea used is the following: An accident is likely to be served late if there are no idle vehicles present at the nearest base. For many EMS regions, for example in the Netherlands, this is typically far from the truth. Moreover, the policies should work well for densely populated areas, the more difficult case of ambulance planning, where some demand points can be reached within the time threshold from as many as 8 different base locations. This complexifies the construction of good base functions.

## 1.2 Our Contribution

In practice, ambulance planners face a number of challenges. Usually only limited and coarse-grained information about the state of the system is available for decision making, while the accuracy of the computations should be good, and at the same the computation times should not be prohibitively large. Motivated by this, the goal of this paper is to propose an algorithm that efficient yet easy-to-use, thereby properly balancing the trade-off between simplicity, accuracy and scalability. Thereby, we ensure that even EMS providers with few tools available to track real-time information, can implement this solution. We focus on busy, urban areas. In such a setting it is unacceptable to move every vehicle each time an accident occurs. And although some pro-active relocations may be useful, they clearly enlarge the workload for the crew. We choose to limit our repositioning opportunities in the following way. An ambulance is only allowed to relocate when it becomes idle (which can be at the incident scene, or at a hospital). Thereby, the number of trips will be the same as for a static strategy, which will help convince EMS managers that our proposed solution is a good alternative to a static strategy.

The ambulance redeployment algorithm we develop in this paper, is both intuitively clear and computable in real time. The solution does not require a preparatory learning phase and is easy to implement. Furthermore, the algorithm requires very little real time data, in fact, only the destinations (locations) of the available vehicles are used. We then decide where to send the available vehicle, based on an expression for marginal coverage improvement. Marginal coverage is an idea that originated in static ambulance planning [5], but this paper shows that it can be useful in dynamic ambulance planning as well. Through this notion of coverage we aim to reduce our KPI: the expected fraction of late arrivals. Our algorithm is designed particularly for busy (urban) areas, but with some adaptations the same technique also works for more rural regions. From a practical perspective, the solution is easily extendable for many restrictions that may occur in real life, e.g., a maximum capacity per base. Since the computation is not a black box, this will help when convincing EMS managers to start using this policy.

Throughout this paper, the key performance indicator (KPI) is the expected fraction of late arrivals. In order to obtain this KPI, we simulate the EMS regions and report the observed fractions of late arrivals - an estimator for the true performance. Our results show that we can obtain an average of 7.8% late arrivals, compared to 9.5% for a benchmark static policy under the same circumstances. In fact, our simulations show that our policy not only performs better for the time threshold, but shifts the entire distribution of response times to the left.

The rest of this paper is structured as follows. In Section 2 we formulate the problem and describe the MEXCLP model in detail. In Section 3 we give our ambulance redeployment algorithm and analyze its computation time. In Section 4 we describe our case studies and measure the performance of our algorithm on these cases. We do a small case study, allowing us to compute the optimal static policy, which we compare to our dynamic policy. We also include a realistic case study on one of the largest EMS regions in the Netherlands. We end with our conclusions in Section 5.

# 2 Problem Formulation

In this section, we introduce the real-time ambulance redeployment problem. To formulate the problem, we define the set $V$ as the set of locations at which demand for ambulances can occur. Note that the demand locations are modeled as a set of discrete points. Accidents at locations in $V$ occur according to a Poisson process with a rate $\lambda$. Let $d_i$ be the fraction of the demand rate $\lambda$ that occurs at node $i$, $i \in V$. Then, on a smaller scale,

| | |
|---|---|
| $A$ | The set of ambulances. |
| $V$ | The set of demand locations. |
| $H$ | The set of hospital locations, $H \subseteq V$. |
| $W$ | The set of base locations, $W \subseteq V$. |
| $T$ | The time threshold. |
| $\lambda$ | Accident rate. |
| $d_i$ | The fraction of demand in $i$, $i \in V$. |
| $\tau_{ij}$ | The driving time between $i$ and $j$ with siren turned on, $i, j \in V$. |
| $n_i$ | The number of idle ambulances that have destination $i$, $i \in W$. |

Table 1: Notation.

accidents occur at node $i$ with rate $d_i\lambda$.

Let $A$ be the set of ambulances. When an accident has occurred, we require the nearest (in time) available ambulance to immediately drive to the scene of the accident. We assume that the travel times $\tau_{ij}$ between two nodes $i, j \in V$ are deterministic. Idle ambulances can only be on the road while driving to a base location in the set $W \subseteq V$, or be at a base location itself waiting for an accident to respond to. Note that idle ambulances on the road may be dispatched immediately, and need not arrive at the base location they were headed to. When an accidents occurs and there are no ambulances idle, the call goes into a first-come first-serve queue. Accidents have the requirement that an ambulance must be present within $T$ time units. When an ambulance arrives at the accident scene, it provides service for a certain random time $\tau_{onscene}$. Then it is decided whether the patient needs transport to a hospital. If not, the ambulance immediately becomes idle. Otherwise, the ambulance drives to the nearest hospital in a set $H \subseteq V$. Upon arrival, the patient is transferred to the emergency department, taking a random time $\tau_{hospital}$, after which the ambulance becomes idle.

We allow an ambulance only to relocate whenever it becomes idle, which could be at the accident scene or at a hospital. Although this choice may seem restrictive, it is a very reasonable choice, and is both crew and fuel friendly. In particular, in complicated busy regions, an ambulance becomes idle quite often. Our restriction on relocation moments provides the system enough freedom to keep updating and avoids getting stuck in a local optimum. In our model, any ambulance is capable of serving any accident. An ambulance is able to respond to an accident (queued or newly arriving), immediately when it becomes idle. Note that this implies that the vehicle does not need to return to a base location before being dispatched again.

## 2.1 State Space and Policy Definition

When defining the state space, one should consider all information of the EMS system that the best relocation might depend on. In a way, the state should represent a 'snap shot' of the system at a decision moment. Most dynamic models (see Section 1.1) use a rather elaborate description of the system, which results in a large state space. In contrast, we will define a relatively small state space, which will help us obtain an intuitive policy that can be understood and explained to EMS employees in practice.

We define the state space as the destinations of all idle ambulances. (If an ambulance is waiting to be dispatched, we say its destination is simply its current location.) It should be clear that this definition of the state space ignores many details of the system, such as information about the busy vehicles and the exact location of ambulances that are driving. Note that ignoring this information (which might affect the best relocation decision) implies that we cannot possibly hope for our method to find an optimal solution. Nevertheless, we show that we can obtain a policy with good performance using only this small state space.

Remember that idle ambulances can only be sent to the predefined base locations in $W$. Furthermore, the vehicles are exchangeable or identical. It is then sufficient to model the state as the number of idle ambulances that are headed to each base location. Hence, define the state space $\mathcal{S}$ to be the set of states $s = \{n_1, \dots, n_{|W|}\}$ such that $n_i \in \mathbb{N}$ for $i = 1, \dots, |W|$ and $\sum_{i=1}^{|W|} n_i \leq |A|$. Here, $n_i$ represents the number of idle ambulances that have destination $i$. We also define the action space $\mathcal{A} = W$, where the action represents the new destination for the newly available ambulance. Now we can define a *policy* $\pi$, as a mapping $\mathcal{S} \to \mathcal{A}$. Let $\Pi$ denote the set of all such policies.

## 2.2 Objective

We look for a relocation policy that minimizes the expected fraction of accidents that are reached later than $T$. Recall that accidents are generated according to the Poisson process described above. Therefore, we can give our accidents an index $i = 1, 2, \ldots$, sorted by their arrival time. Now we can express our objective as:

$$\arg\min_{\pi \in \Pi} \lim_{I \to \infty} \frac{\sum_{i=1}^{I} \mathbb{1}[h^{\pi}(i) - t(i) > T]}{I}, \tag{1}$$

where $t(i)$ represents the time that accident $i$ occurs, and $h^{\pi}(i)$ represents the time a vehicle arrives at the scene of accident $i$, under policy $\pi$.

Our model uses two different travel speeds. If the ambulance is traveling without siren, its travel speed is 0.9 times the travel speed when it is traveling towards an accident scene.

# 3 Algorithm

In this section, we develop an algorithm to solve the dynamic ambulance relocation problem. Our goal is to minimize the expected fraction of late arrivals. In order to reach this goal, we will use the notion of coverage. It is intuitive that a well-covered region will result in a small expected fraction of late arrivals. Thereto, we can benefit from a related coverage model that we will describe next.

## 3.1 A Related Model

We highlight a related model called the maximum expected covering location problem formulation (MEXCLP) [5]. This is a model that searches for the best *static* policy using integer linear programming. Although static models are conceptually different from the dynamic policy that we are looking for, the underlying idea of MEXCLP will turn out to be useful.

In this formulation there is a limited number, say $|A|$, ambulances that need to be distributed over a set of possible base locations $W$. Each ambulance is modeled to be unavailable with a pre-determined probability $q$, called the *busy fraction*. Note that it is implicitly assumed that this probability is the same for all vehicles, regardless of their position with respect to the demand and the other vehicles. The busy fraction can be estimated by dividing the expected load of the system by the total number of available ambulances. Consider a node $i \in V$ that is within range of $k$ ambulances. The travel times $\tau_{ij}$, $i, j \in V$ are assumed to be deterministic, which allow us to straightforwardly determine this number $k$. If we let $d_i$ be the demand at node $i$, the expected covered demand of this vertex is $E_k = d_i(1 - q^k)$. The authors of [5] show that the marginal contribution of the $k$th ambulance to this expected value is $E_k - E_{k-1} = d_i(1-q)q^{k-1}$. We introduce a binary variable $y_{ik}$ that is equal to 1 if and only if vertex $i \in V$ is within range of at least $k$ ambulances. The variables $x_j$ (for $j \in W$) represent the number of vehicles at each base. Let $W_i$ denote the set of bases that are within range of demand point $i$, that is: $W_i = \{j \in W : \tau_{ij} \leq T\}$, then we can formulate the MEXCLP model as:

$$\text{Maximize } \sum_{i \in V} \sum_{k=1}^{p} d_i(1-q)q^{k-1} y_{ik}$$

subject to

$$\sum_{j \in W_i} x_j \geq \sum_{k=1}^{p} y_{ik}, \quad i \in V,$$

$$\sum_{j \in W} x_j \leq |A|,$$

$$x_j \in \mathbb{N}, \quad j \in W,$$

$$y_{ik} \in \{0, 1\}, \quad i \in V, k = 1, \ldots, p.$$

Note that there is no need to add the extra constraint $y_{ih} \leq y_{ik}$ for $h \leq k$. This will always hold for an optimal solution, since $E_k - E_{k-1}$ is decreasing in $k$.

In Section 3.2, we reuse the MEXCLP expression for the marginal coverage contribution $(E_k - E_{k-1})$ to obtain a dynamic redeployment strategy.

## 3.2 Algorithm Description

Our aim is to use as little information possible, such that it can be applied in very general settings, and such that it is implicitly insensitive to changes or estimation errors of the parameters. Hence, we search for a redeployment policy $\pi$, using the state space as described in Section 2. This means that whenever an ambulance becomes idle, we can only use the destinations of all other idle ambulances to base our decision on. This corresponds to taking a decision in the state in which all idle ambulances have arrived at their destination. Note, however, that this situation may not even occur, because accidents may occur or other vehicles may become idle in the mean time. However, it will turn out to be a useful state description nonetheless.

Recall that we are looking for a policy that minimizes the expected fraction of late arrivals over a set of random accidents (see Equation (1)). At any decision moment, the idle ambulances at that epoch already provide a certain coverage of the region. We then decide where to send the vehicle that is about to become idle, by calculating the coverage improvement when it is sent to base $w$, for all $w \in W$. Note that there are several definitions of 'coverage', which all lead to different redeployment strategies. We find it instructive to first address the most basic notion of coverage. This results in a myopic redeployment policy. We discuss its behavior and shortcomings, which builds up to our proposed solution that uses the same definition of coverage as the MEXCLP model.

### Myopic Solution

At decision moments, we can straightforwardly calculate which regions are not covered at all. That is, the demand nodes that are further than $T$ away from any idle ambulance destination. We can then make a greedy choice by sending the newly idle ambulance to a base that covers most of the yet uncovered demand. Note that this is a myopic solution, it is in fact a dynamic version of the Maximum Coverage Location Problem (MCLP) [4]. We have implemented this policy, and found that its performance hardly improved the static MEXCLP solution. (For some choices for the parameters of the system, the performance was even worse than the static solution.) The intuition is that this policy steers towards a configuration that is optimal with respect to covering the next emergency call, but it lacks the insight of how much coverage is left after responding to the first call. This is typical for myopic policies, and in order to overcome this, we require some quantification of where there will be a shortage of ambulances in the future.

### Dynamic MEXCLP Solution

In order to obtain a good policy, we need to include some measure of how much coverage we can provide in the future. In other words, we need to take into account that some of the currently idle vehicles may be dispatched, and ensure the remaining coverage in the future is still good. Therefore, we propose a policy that sends the idle ambulance to the base that results in the largest marginal coverage according to the MEXCLP model. This describes the benefit of adding a $k^{th}$ ambulance within range of demand node $i$. Recall that this is given by $E_k - E_{k-1} = d_i(1-q)q^{k-1}$. We choose the base that gives the largest marginal coverage over all demand, which implies that also the largest coverage overall is obtained. This can be expressed as follows.

$$\pi(\{n_1, \ldots, n_{|W|}\}) = \arg\max_{w \in W} \sum_{i \in V} d_i(1-q)q^{k(i,w,n_1,\ldots,n_{|W|})-1},$$

$$\text{where } k(i, w, n_1, \ldots, n_{|W|}) = \sum_{j=1}^{|W|} n_j \cdot \mathbb{1}(\tau_{ji} \leq T) + \mathbb{1}(\tau_{wi} \leq T).$$

Here, $\mathbb{1}$ denotes the indicator function. The travel times $\tau_{ji}$ are taken as estimates for movements with siren turned on. We perform the search for the best relocation brute force, as described in Algorithm 1.

```
Data: The demand $d_i$ per node $i \in V$,
base locations $W \subseteq V$,
busy fraction $q \in [0,1]$,
current destinations $dest(a)$ for all $a \in IdleAmbulances \subseteq A$
travel times $\tau_{ij}$ between any $i, j \in V$,
time threshold $T$ to reach an emergency call.
Result: A new destination for the ambulance that is about to become
        idle
BestImprovement = 0
BestLocation = NULL
foreach $j$ in $W$ do
    CoverageImprovement = 0
    foreach $i$ in $V$ do
        $k = 0$
        if $\tau_{ji} \leq T$ then
            $k$++
            foreach $a$ in IdleAmbulances do
                if $\tau_{dest(a)i} \leq T$ then
                    $k$++
                end
            end
            CoverageImprovement $+=$ $d_i(1-q)q^{k-1}$
        end
    end
    if CoverageImprovement > BestImprovement then
        BestLocation = $j$
        BestImprovement = CoverageImprovement
    end
end
return BestLocation
```

**Algorithm 1:** Dynamic MEXCLP

## 3.3 Limitations

As described in Section 2.1, our state space definition prohibits the ambulance relocation problem from being solved to optimality. But even within our state space, the Dynamic MEXLP model need not lead to optimal decisions. The definition of (marginal) coverage as given by the MEXCLP model has some well-known imperfections. For example, vehicles are assumed to operate independently, and the busy fraction is assumed to be the same for all vehicles. These limitations also transfer to the dynamic usage of (MEXCLP) coverage. Therefore, our proposed solution must be a heuristic one, and we do not claim to have solved the problem in an exact manner. However, heuristic policies are common in dynamic ambulance planning, due to the difficulty of the problem. Furthermore, we consider the MEXCLP definition of coverage an elegant one, and it allows for fast computations (as we will see in Section 3.4).

### 3.4 Computation Time

We analyse the computation time of dynamic MEXCLP, in order to determine the scalability of our method. In Algorithm 1 it is easy to see that we loop over all bases, demand nodes and idle ambulances. Therefore, the dynamic MEXCLP algorithm runs in $\mathcal{O}(|W||V||A|)$ iterations.

In practice the number of base locations is typically small, e.g., 20 or 30. Also the number of ambulances that an EMS provider uses, is very limited. The size of $V$ is mostly dependent on the way the data is aggregated, and it is the only quantity that is likely to be large. The fact that the computation time is linear in $|V|$, ensures that Algorithm 1 will remain tractable even for large regions or regions with a high level of detail.

## 4 Computational Results

In this section we verify our dynamic MEXCLP repositioning policy by simulating several EMS regions. To this end, we built a discrete event simulation model that keeps track of all accidents and vehicles. There are events for an accident occurring, an ambulance arriving at the scene of the accident, an ambulance leaving for a hospital, an ambulance arriving at a hospital, and an ambulance becoming idle.

When an accident occurs, the closest idle ambulance is dispatched. For every vehicle we keep track of the origin and destination, including the start time of its movement. This allows us to determine where moving ambulances are while we look for the closest available vehicle. We do this by a linear interpolation between the origin and destination, given the time since the ambulance started moving and the known total driving time from origin to destination. We then round our result down to the nearest point in $V$, since our estimates for driving times are only given between points in $V$. Our experiments show that for the majority of the accidents, approximately 77%, the corresponding ambulance departs from a base location.

When an ambulance completes an accident, we check if there are any unattended accidents left in the queue. If not, the ambulance becomes idle, and is sent to a base location[2]. In our proposed solution, this base location is determined by Algorithm 1. As benchmarks, we use so-called static solutions, in which the idle ambulance returns to its own pre-defined home base. This is a typical benchmark in ambulance redeployment literature (used, e.g., in [13] and [20]).

We measure the fraction of ambulances arriving at the scene of an accident with a response time larger than $T$.

## 4.1 A Small Region

We first start with a tractable region, which consists of a small number of demand nodes. This is insightful as it allows for a brute force search among all static policies. For a more realistic case study, we refer the reader to Section 4.2.

The region we use is inspired by a small part of the Netherlands. We aggregate the demand on the level of municipalities, which in this case boils down to cities and towns. Furthermore, we add three nodes, A, B and C, that are located at important road intersections. These last nodes have no demand, but it is possible to strategically station an ambulance there. For the geographical characteristics of the region, see Figure 1. In this region there is only one hospital, which is located in City 2.

For illustration, we set the time threshold to $T = 10$ minutes, and use demand as described in Table 2. Furthermore, we allow exactly 5 ambulances to serve the accidents in this region.

### Static Policies

Let us consider static policies first. We have 9 nodes and 5 vehicles available. If vehicles were distinguishable, this would mean there are $9^5 = 59,049$

[2] Recall that the ambulance might not arrive at this base location, because it may be dispatched before reaching its destination.

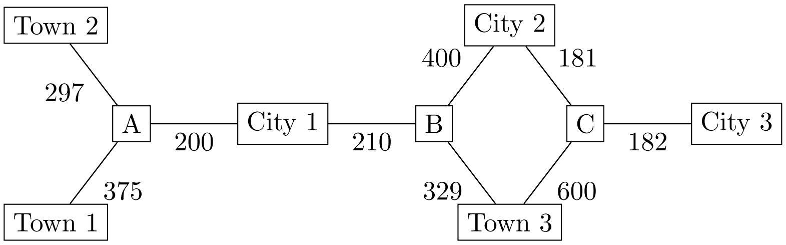


Figure 1: A graph representation of the region. The numbers on the edges represent the driving times in seconds with siren turned on.

| $i$ | $d_i$ |
|---|---|
| City 1 | 0.2 |
| City 2 | 0.4 |
| City 3 | 0.2 |
| Town 1 | 0.07 |
| Town 2 | 0.07 |
| Town 3 | 0.06 |
| A | 0 |
| B | 0 |
| C | 0 |

Table 2: Distribution of demand in a small region

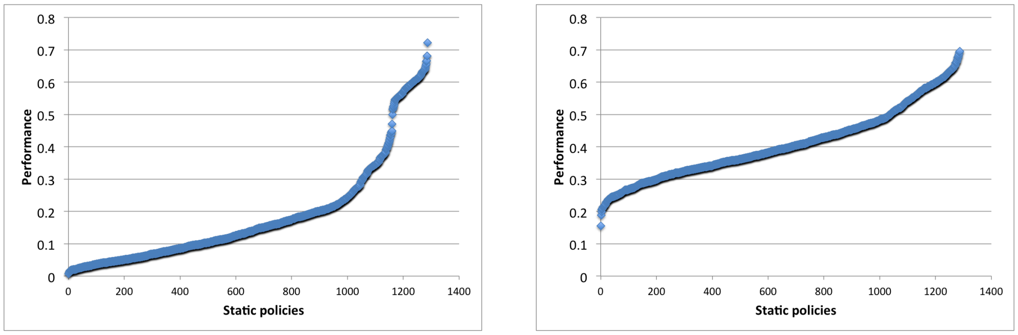
different static policies. Instead, we assume vehicles are indistinguishable, which makes the set of truly different policies smaller. If we number the nodes 1 up until 9, we can describe a policy by a five tuple of non-decreasing integers, representing the home locations of the five vehicles. E.g., (2,2,5,8,9) denotes a policy, but (5,6,3,1,9) does not. Using this definition, we can iterate over all static policies. This allows us to take a closer look at the static solution space. Finding the optimal solution for a discrete event dynamic system (DEDS) is in general difficult due to the large search space and the simulation-based performance evaluation. Inspired by Ordinal Optimization (see, for example, [11] or [16]), which has become an important tool for optimizing DEDSs, we create an Ordered Performance Curve (OPC) as follows. For each policy, we simulate the EMS region for an amount of time, and use the measured fraction of late arrivals as an estimate for the true performance of the policy.[3] Then, we sort the policies by their estimated performance, giving us the desired OPC. At first, we look into the case where there are relatively few accidents, i.e., $\lambda = 1/45$ minutes. In this case, we evaluate each policy with 10 simulated days. For the corresponding OPC, see Figure 2a. According to the theory of Ordinal Optimization, the shape of this OPC indicates that there are many good solutions (policies) for this problem.

However, it would be incorrect to conclude that this is true for all static ambulance positioning problems. In fact, our experiments show that changing the accident rate $\lambda$, while keeping all other parameters the same, already affects the shape of the OPC. For $\lambda = 1/13$ minutes, the OPC is shown in Figure 2b. For this case, we evaluate each policy with 2.9 simulated days, which boils down to the same expected number of accidents per evaluation as in the $\lambda = 1/45$ case. First of all, note that the best static solution for this problem seems to have a performance of 17% (compared to 1% in Figure 2a). An increase was to be expected, because the same number of vehicles needs to serve a higher number of accidents. Perhaps more surprising is that also the shape of the OPC has changed. For Figure 2b, the OPC indicates that there exist only a few good static policies for this problem.

In order to determine the best static policy, we perform longer simulations to explore the region of the good solutions with more accuracy. Note that when $\lambda$ changes, the optimal static policy may change as well. In fact, we

[3] We start with an empty system, i.e., no accidents have occurred. Therefore, we need to allow the system some time to evolve towards a more natural and representative state. We disregard the first five simulated hours in each run, and only consider the performance of the remaining time.

find that for $\lambda = 1/45$ the best static policy is (City 1, City 1, City 2, C, C), while for $\lambda = 1/13$ the best static policy is (City 1, City 1, City 2, City 2, C).



(a) $\lambda = 1/45$ minutes

(b) $\lambda = 1/13$ minutes

Figure 2: OPC curves for static policies in the same region, for two different accident intensities.

### DMEXCLP versus the Best Static Policy

We now compare the performance of dynamic MEXCLP (DMEXCLP) with the best static policy. We will test our method on multiple scenarios, to show that the method gives good results for more than just one specific problem instance. We create different problem instances by changing the value of $\lambda$. Since we keep the number of vehicles equal to 5, by varying $\lambda$ we also vary the load of the system. In Figure 3, it shows that the DMEXCLP policy outperforms the best static policy for every choice of $\lambda$. When we let $\lambda$ take even more extreme values, we see that DMEXCLP has approximately the same performance as the best static solution. This occurs when $\lambda = 1/9$ minutes, in which case the expected fraction of late arrivals for both the best static and the DMEXCLP solution is around 67%. A fraction this high will never be acceptable in real life, and would indicate that more vehicles are needed. Therefore, we should not draw conclusions on the applicability based on this parameter choice. Note that, even if the performance of DMEXCLP is equal to the performance of the best static policy, DMEXCLP is still useful in the sense that its calculations are faster than the search for the best static policy.

In Figure 4 we see that the relative performance improvement for this region can be as high as 20%. In the following section we will investigate whether this number is representative for a more realistic region with demand aggregated on a smaller scale.

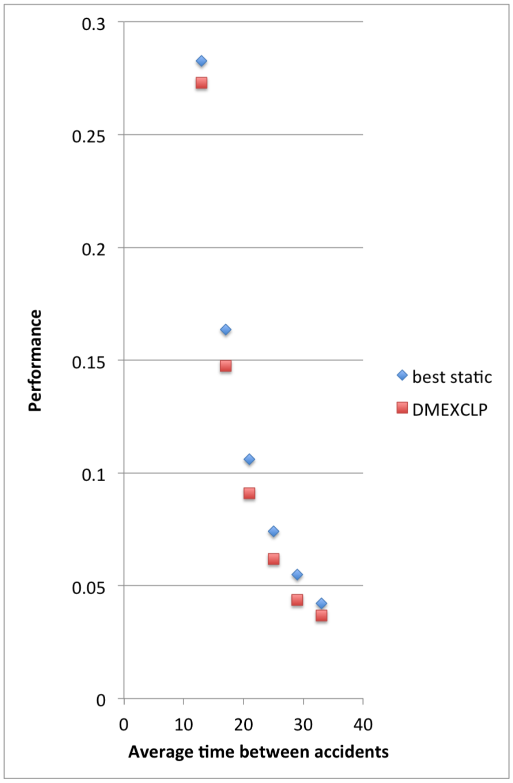


Figure 3: The absolute performance (expected fraction of late arrivals) of Dynamic MEXCLP compared to the best static policy. The horizontal axis displays the average time between accidents in minutes. Each policy was evaluated long enough such that the tolerance interval (1.96 times the sample standard deviation) is within 2.5% of our estimated value.

## 4.2 A Realistic Case Study

In this section, we validate our redeployment method on a realistic problem instance. We chose to model the region of Utrecht, which is one of the largest ambulance providers of the Netherlands. For the parameters used in

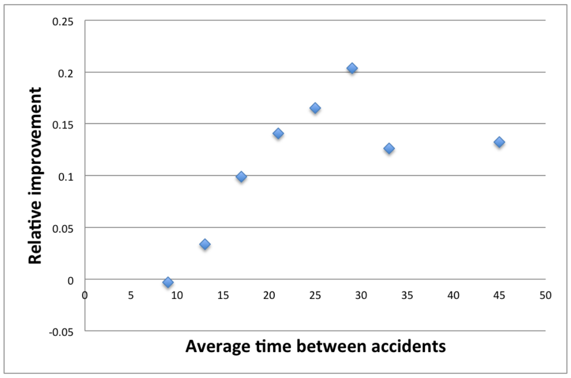


Figure 4: The relative improvement in performance of Dynamic MEXCLP compared to the best static policy. The horizontal axis displays the average time between accidents in minutes. Each policy was evaluated long enough such that the tolerance interval (1.96 times the sample standard deviation) of both policies is within 2.5% of the estimated value.

the implementation, see Table 3. This is a region with multiple hospitals, and for simplicity we assume that the patient is always transported to the nearest hospital, if necessary.

Note that we use the fraction of inhabitants as our choice for $d_i$. In reality, the fraction of demand could differ from the fraction of inhabitants. However, the number of inhabitants are known with great accuracy, and this is a straightforward way to obtain a realistic setting. Furthermore, the analysis of robust optimization for uncertain ambulance demand in [9] indicates that we are likely to find good solutions, even if we make mistakes in our estimates for $d_i$.

In the Netherlands, the time target for the highest priority emergency calls is 15 minutes. Usually, 3 minutes are reserved for answering the call, therefore we choose to run our simulations with $T = 12$ minutes. The driving times for EMS vehicles between any two nodes in $V$ were estimated by the Dutch National Institute for Public Health and the Environment (RIVM) in 2009. These are driving times with the siren turned on. For ambulance movements without siren, e.g., when repositioning, we use 0.9 times the speed with siren. The number of vehicles used in our implementation is such that a good policy gives a performance (expected fraction of late arrivals) of a magnitude that is realistic for practical purposes.

| parameter | magnitude | choice |
|---|---|---|
| $\lambda$ | 1/9.5 minutes | Realistic for urgent calls on a weekday in this region. |
| $A$ | 19 | Realistic number to cover demand. |
| $W$ | 19 | Base locations as existing in 2013. |
| $V$ | 217 | 4 digit postal codes. |
| $H$ | 10 | The hospitals within the region in 2013, excluding private clinics. |
| $\tau_{ij}$ | | Driving times as estimated by the RIVM. |
| $d_i$ | | Fraction of inhabitants as known in 2009. |

Table 3: Parameter choices for our implementation of the region of Utrecht.

### Results

We compare the performance of the dynamic MEXCLP solution with a benchmark. We let the benchmark be the static MEXCLP solution, which is generally assumed to give a good static policy (for a comparison of static methods, see [18]). Note that the verification of the value of one single policy is not feasible within polynomial time. Therefore, it is not tractable to perform a brute force search over all static policies using 19 base locations and 19 vehicles. Since there is no alternative known to compute the optimal static solution, this means we cannot use the optimal static solution as a benchmark.

In both the static (benchmark) and the dynamic (proposed solution) case, we initialize the locations of the ambulances according to the static MEXCLP solution. We simulate the EMS system 10 times per policy and compare the results in Figure 5. We measure the fraction of late arrivals, which decreased from on average 9.5% to 7.9%. This is a difference of 1.6 percentage point, and a decrease of 16.8%. This is a significant improvement that can be made without purchasing extra vehicles or increasing the number of crew shifts. Furthermore, this improvement is large in comparison to other results in literature (e.g., an improvement from 26.7% to 25.8% in [12], which boils down to a 3.4% gain).

We would like to emphasize that the dynamic MEXCLP policy does not

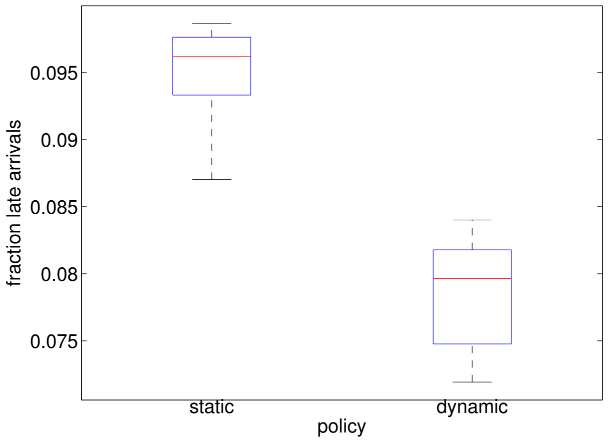


Figure 5: Comparing the performance of Dynamic MEXCLP with the static MEXCLP solution. For both policies a value of $q = 0.3$ is used. Each policy was evaluated with 10 runs of 500 simulated hours.

only reduce the expected fraction of late arrivals, but also reduces the average response times overall. This can be concluded from Figure 6.

### 4.3 Sensitivity to the Busy Fraction

We investigate the sensitivity of Algorithm 1 to the parameter $q$, the busy fraction. In order to do this, we keep the number of vehicles equal to 19, and we also keep the average time between accidents equal to 9.5 minutes. We run the DMEXCLP algorithm for several values of $q$, and compare the performance in Figure 7. We conclude that, at least for this particular problem instance, the quality of the solution is very insensitive to the value of parameter $q$.

## 5 Conclusions

In this paper we have developed real-time scalable algorithms for dynamic ambulance redeployment with a focus on minimizing the expected fraction of late arrivals. We have introduced a dynamic MEXCLP heuristic (see Algorithm 1) that reduces the expected fraction of late arrivals by relatively

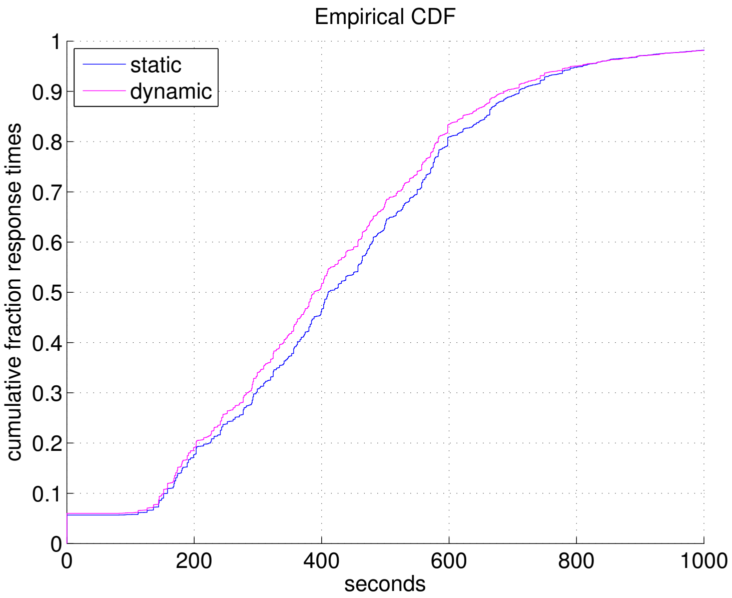


Figure 6: Response times for dynamic MEXCLP and the static MEXCLP solution. For both policies a value of $q = 0.3$ is used. Each policy was evaluated with 2,500 simulated hours.

16.8% compared to a good static policy. Additionally, the dynamic MEXCLP heuristic also reduces the average response times overall. The heuristic depends on the busy fraction, i.e., the fraction of time that an ambulance is unavailable, that needs to be estimated. Our experiments indicate that good performance is still obtained, even if there is an error in the estimation of the busy fraction.

## 5.1 Remarks

In terms of applicability, we find it useful to consider whether the Dynamic MEXCLP heuristic is still feasible when we relax some of our assumptions. We address the following cases.

### Changes During the Day

In practice, EMS systems may deal with characteristics that change over the course of a day. This is reflected in changing parameters in our model. We

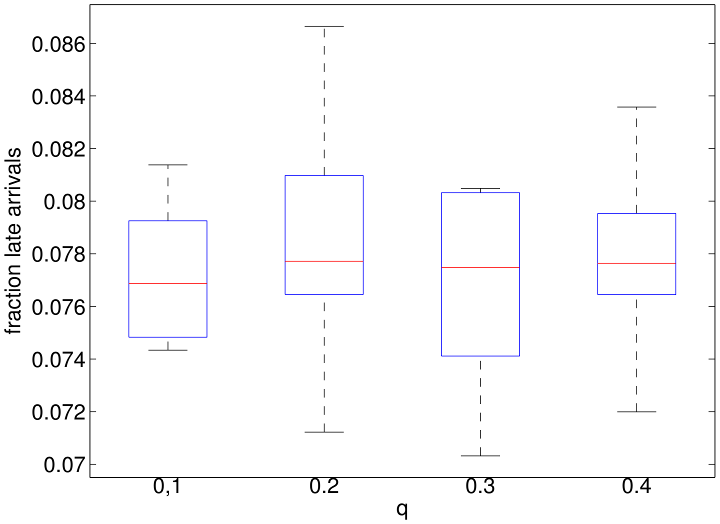


Figure 7: Comparing the performance of DMEXCLP for several values of $q$. The boxes consist of ten runs, in which we simulate 1000 hours, each.

mention a few examples.

- Accident probabilities may shift, for example, an accident is more likely to occur in an industrial area during office hours.
- Travel times may be longer in rush hour, or may depend on the weather.

Changing parameters over time, such as the examples above, are often difficult to incorporate in a solution. However, in our case, there is no need to complicate the algorithm. At any decision epoch, use the parameters that are relevant for the upcoming period. The choice of the period size may depend on the EMS region, but for example 30 minutes would be a good starting point.

However, we want to point out that emergency services do not always experience the impact of the time of day on their response velocities. For example, empirical evidence shows only a minor impact for fire fighters in New York [10] and ambulances in Calgary [2]. Furthermore, even if one is certain that the time of day is relevant for the response velocities, the task remains to estimate the different velocities accurately. Care has to be taken as to not make mistakes, e.g., due to the data containing only a small number of trips from $i$ to $j$ in each time segment. At this moment, we do not have access to accurate time dependent travel time estimates, and therefore we did not implement such a case study.

### Stochastic Travel Times

One straightforward way of dealing with stochastic travel times, is to use the expectations $E[\tau_{ij}]$ in Algorithm 1. Alternatively, we did some additional simulations, in which we found good performance when using the 0.8 quantile, i.e., the number $X_{ij}$ such that $P[\tau_{ij} \leq X_{ij}] = 0.8$. The performance will of course depend on the exact distribution function chosen, and we suggest some preliminary experiments to obtain a good strategy.

### Staff Satisfaction

Staff members that come from a 'static' work environment may be used to having their own, fixed home base. Giving up this concept can be difficult. Although our proposed method already limits the relocation moments, extra adjustments can be made to accommodate the staff. For example, a good compromise would be the following. Each vehicle (and the corresponding crew) still has its own, fixed home base. Preferably, we send the vehicle to this home base, but we may choose another base if the expected gain is large enough. One can measure this by calculating the marginal coverage that would be obtained if we were to send the vehicle to its own home base, and compare this with the marginal coverage that could be obtained by a relocation. Finally, one might relocate the vehicle if and only if the difference in marginal coverage is greater than a certain threshold.

### Rural Regions

As we mentioned in Section 1.2, our algorithm is designed particularly for busy (urban) areas. For rural regions, however, the same technique may still be applicable, albeit with some adaptations. A key observation is that rural regions have a lower accident frequency - which is directly related to the frequency at which ambulances become idle. This implies that there will be fewer relocation moments, and therefore we expect performance improvements to be smaller. In order to overcome this, we suggest adding some additional relocations[4]. For example, one could allow a relocation when a new accident arrives. In addition, it is possible to allow *two* vehicles to relocate upon completion of an accident. The decision on where to send the vehicles, can still be made using the Dynamic MEXCLP method.

### Multiple Targets

In some countries there exist multiple time targets, depending on the urgency of the situation. For example, in the Netherlands, the highest priority accidents have to be reached within 15 minutes, and the less severe (but still urgent) accidents have to be reached within 30 minutes. We advise to apply the Dynamic MEXCLP algorithm using the most strict time target. Our numerical experiments regarding realistic use cases, indicate that this results in a policy that also has a good performance for a target of 30 minutes.

## Acknowledgements

The authors of this paper would like to thank the Dutch Public Ministry of Health (RIVM) for giving access to the travel times for EMS vehicles in the Netherlands. We also thank the editor and the anonymous referee for their useful comments. The first author thanks the EMS provider of the region Utrecht (Ravu), for discussions, and the opportunity to experience several aspects of the EMS system in practice. This research was financed in part by Technology Foundation STW under contract 11986, which we gratefully acknowledge.

## References

[1] R. Alanis, A. Ingolfsson, and B. Kolfal. A Markov chain model for an EMS system with repositioning. *Production and Operations Management*, 22(1):216–231, 2013.

[2] S. Budge, A. Ingolfsson, and D. Zerom. Empirical analysis of ambulance travel times: The case of Calgary emergency medical services. *Management Science*, 56(4):716–723, 2010.

[4] This will obviously increase the workload for the crew, but we think this is acceptable since a rural region is typically not very busy.

[3] G. Cady. JEMS 200 city survey, JEMS 2001 annual report on EMS operational & clinical trends in large, urban areas. *JEMS: A Journal of Emergency Medical Serices*, 27(2):46–71, 2002.

[4] R.L. Church and C.S. Revelle. The maximal covering location problem. *Papers of the Regional Science Association*, 32:101–118, 1974.

[5] M.S. Daskin. A maximum expected location model: Formulation, properties and heuristic solution. *Transportation Science*, 7:48–70, 1983.

[6] M. Gendreau, G. Laporte, and F. Semet. A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. *Parallel Computing*, 27:1641–1653, 2001.

[7] J. Goldberg, R. Dietrich, J.M. Chen, and M.G. Mitwasi. Validating and applying a model for locating emergency medical services in Tucson, AZ. *Euro*, 34:308–324, 1990.

[8] K. Hogan and C.S. Revelle. Concepts and applications of backup coverage. *Management Science*, 34:1434–1444, 1986.

[9] R.B.O. Kerkkamp. Optimising the deployment of emergency medical services. Master's thesis, Delft University of Technology, 2014.

[10] P. Koleskar, W. Walker, and J. Hausner. Determining the relation between fire engine travel times and travel distances in New York City. *Operations Research*, 23(4):614–627, 1975.

[11] L. Lee, T. Lau, and Y. Ho. Explanation of goal softening in ordinal optimization. *IEEE Transactions on Automatic Control*, 44(1):94–99, 1999.

[12] M.S. Maxwell, S.G. Henderson, and H. Topaloglu. Tuning approximate dynamic programming policies for ambulance redeployment via direct search. *Stochastic Systems*, 9999. To appear.

[13] M.S. Maxwell, M. Restrepo, S.G. Henderson, and H. Topaloglu. Approximate dynamic programming for ambulance redeployment. *INFORMS Journal on Computing*, 22:226–281, 2010.

[14] W. B. Powell. *Approximate Dynamic Programming: Solving the Curses of Dimensionality.* John Wiley & Sons, 2011.

[15] J.F. Repede and J.J. Bernardo. Developing and validating a decision support system for locating emergency medical vehicles in Louisville, Kentucky. *European Journal of Operational Research*, 75:567–581, 1994.

[16] Z. Shen, Q.-C. Zhao, and Q.-S. Jia. Quantifying heuristics in the ordinal optimization framework. *Discrete Event Dynamic Systems*, 20(4):441–471, 2010.

[17] C. Toregas, R. Swain, C. ReVelle, and L. Bergman. The location of emergency service facilities. *Operations Research*, 19(6):1363–1373, 1971.

[18] P.L. van den Berg, J.T. van Essen, and E.J. Harderwijk. Comparison of static ambulance location models. *Under review*, 2014.

[19] D.M. Williams. 2008 JEMS 200 city survey: The future is your choice. *JEMS: A Journal of Emergency Medical Services*, 34(2):36–51, 2009.

[20] Yisong Yue, Lavanya Marla, and Ramayya Krishnan. An efficient simulation-based approach to ambulance fleet allocation and dynamic redeployment. In *AAAI Conference on Artificial Intelligence (AAAI)*, July 2012.

[21] O. Zhang, A.J. Mason, and A.B. Philpott. *Simulation and optimisation for ambulance logistics and relocation.* Presented at the INFORMS 2008 Conference, 2008.