A Simulation Package for Emergency Medical Services

S. Ridler, A. J. Mason, A. Raith
Department of Engineering Science
University of Auckland
New Zealand

Abstract

Emergency medical services (EMS) aim to provide timely medical care, and transport of patients. The provision of this service entails many decisions to be made, such as where to place ambulance stations, how many ambulances will be required at each station, ambulance dispatching behaviour, etc. We have developed a simulation package in the programming language Julia, allowing for performance evaluation of different decisions or policies for problems such as these. The simulation model of this package is detailed, along with an example of how the simulation may be used in an optimisation framework.

1 Introduction

The purpose of Emergency Medical Services (EMS) is to respond to medical emergencies, providing timely medical care and, if needed, transport for patients to hospital or other care facilities. EMS providers are commonly required to maintain a minimum level of performance, but have limited resources (staff, EMS vehicles, capital), and so require intelligent decision-making in order to best utilise these resources. Decision-makers for EMS have many problems to solve, such as where to place ambulance stations, how many ambulances or other vehicles will be required at each station, staff scheduling, ambulance dispatching behaviour, etc. In order to evaluate the performance of various system set-ups and policies, a model of the system can be used, rather than experimenting with the real world EMS system which would disrupt usual ambulance operation and could lead to costly mistakes.

For complex, stochastic systems such as the EMS system, it is not possible to create an exactly accurate analytical model, so approximate models are required. One such approximate model is computer simulation, which, if created correctly, can give accurate performance estimates for various system set-ups. Other models include Markov chain models, such as the hypercube queueing model (Larson and Odoni 1981) which calculates performance metrics such as individual ambulance workloads, average travel times, and dispatch frequencies, but this model cannot be used when the dispatch policy is based on emergency priorities. The maximum expected coverage location problem (MEXCLP) (Daskin 1983) is an integer programming model of the expected coverage (a proxy for expected number of calls responded to on time), and it assumes that all ambulances have equal probability of being busy at any moment. An advantage of using a simulation model over an analytical model is that any simplifying assumptions from the analytical model can be removed. Also, results from simulation can include distributions instead of single values such as expectations which are common outputs from analytical models. For evaluating and analysing the performance of EMS operations, simulation is common practice in the literature. A review of EMS simulation models is presented by (Aboueljinane, Sahin, and Jemai 2013). Simulation can also be used in an optimisation framework in order to evaluate an objective function, giving rise to simulation-based optimisation. The use of optimisation in EMS operations can improve resource utilisation and help in reaching or maintaining a minimum level of performance.

Researchers in the field of EMS decision-making typically use different simulation tools and there is no consensus on the most appropriate model choice. This leads to results that cannot be directly compared. For example, one problem studied in the literature is dynamic ambulance redeployment, which is the real-time reassignment of ambulances to stations in order to cover for currently busy ambulances and so improve EMS performance. Many different redeployment models have been created, and the performance of most them has been evaluated with simulation, though different simulation models were used, so the relative performance of each redeployment model is not known. This EMS simulation package has been created in order to enable such a comparison.

This paper first gives an outline of EMS operations in Section 2, detailing the response process for emergency calls. Section 3 gives an overview of EMS simulation models from the literature. The EMS simulation model we have created is presented in Section 4, along with the assumptions, verification and validation, and the open-source software implementation written in the programming language Julia. To demonstrate the value of the simulation package, a simulation-based optimisation example is given in Section 5, before concluding the paper.

2 EMS operation

A common set of steps is followed by EMS providers when responding to emergency calls. Figure 1 is a flowchart summarising the order of operations, with rectangular blocks representing discrete events, and rounded blocks representing decisions. First, the emergency call is received at a call centre and the location and priority of the emergency is determined, e.g. high, medium, or low priority. Once the call has been screened, a dispatcher will decide which ambulance (if any) will respond, ensuring that the ambulance chosen has the correct equipment and appropriately trained staff in order to provide the medical care required. It is common for the nearest idle ambulance to be dispatched, unless all ambulances are busy in which case the emergency is queued (or sometimes a fire response unit may act as a first responder). If the dispatched ambulance is at a station, it may encounter a mobilisation delay due to the time taken for the staff to leave the station. For higher priority emergencies, the ambulance may use lights and sirens in order to pass through road traffic and reach the emergency in a timely manner. Once the ambulance arrives at the emergency location, the condition of the patient is assessed and treatment may begin at the location. After a period of treatment, the patient may require transport to a hospital where they will be handed over to hospital staff for further care. Once an ambulance becomes idle after such a mission it will either be dispatched to another emergency or it will return to a station. If a static ambulance deployment policy is being used, each ambulance would be assigned a home station and it would return to this station when idle.

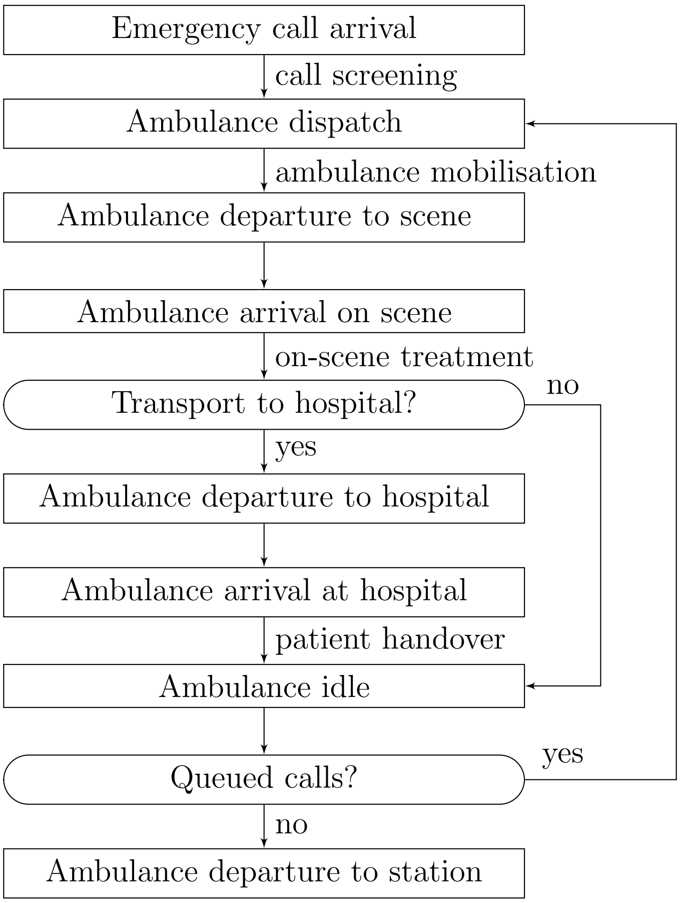

Figure 1: Emergency response flowchart

Other more complex processes can also be followed. For lower priority emergencies, some EMS operators do not always dispatch the nearest idle ambulance; instead, any of the ambulances that can respond to the emergency in time can be considered for dispatching (Mason 2013). Diversions can also be used, in which an ambulance that is en route to a low priority emergency may be diverted to a new high priority emergency if the ambulance is the closest to respond to the new emergency (Mason 2013).

Simulation can model these complex behaviours that other approaches cannot account for, and so it can be a useful tool for accurately evaluating, diagnosing, and improving upon various EMS set-ups.

3 EMS simulation literature review

This section gives a brief overview of EMS simulation model features from the literature. For a more in depth literature review of EMS simulation models, see (Aboueljinane, Sahin, and Jemai 2013), and the literature reviews in (Pinto, Silva, and Young

2015), (Kergosien et al. 2015). Almost all of the simulation models we have seen use discrete event simulation, although other approaches have been used, such as a hybrid of discrete event simulation and agent based simulation (Aringhieri 2010), and a simulation using constant time-steps (Andersson and Värbrand 2007). EMS simulation models have been created and applied to many regions, including cities in New Zealand, Thailand, Taiwan, Brazil, Italy, France, China, Japan, etc. (Pinto, Silva, and Young 2015). The following is a list of common EMS simulation model features:

- Ambulances (and their staff); models typically use homogeneous ambulances for simplicity (for creating the simulation, optimising, and interpreting results), though some use two (or more) classes of ambulances such as advanced and basic life support. Ambulances are modelled as either being available 24 hours each day, or as following shifts; the use of either depends on the study being performed.
- Emergency calls; data for these can be historical, but such data is not always available to researchers, so the use of artificially generated call data is common. There can be one or more emergency priorities; different priorities can have different performance targets, and can lead to behaviour such as ambulance diversion. Emergency arrival rates are commonly modelled with a Poisson point process (homogeneous or inhomogeneous), and service related times (on-scene time, patient hand-over time at hospital) are modelled with a variety of distributions; see Table 1 of (Kergosien et al. 2015).
- Road network; this may have deterministic or stochastic travel times, and these times may be temporally static or dynamic. Some models do not use a road network but instead approximate travel times between locations from a distribution or using a combination of a fixed speed and a simple distance metric (Manhattan or Euclidean).
- Stations; a maximum ambulance holding capacity is usually enforced.
- Hospitals; these are commonly treated as being homogeneous and able to treat any patient, though some models include the requirement that certain patients can only be admitted to certain hospitals in order to model specialised facilities such as burns centres.
- Dispatch logic; typically a nearest-idle ambulance dispatch policy is used. If a queue of emergencies forms then a decision needs to be made as to how the currently busy ambulances will be dispatched as they become available, e.g. dispatch to highest priority emergencies first, in order of occurrence.

The simulation model in our package follows many of the common modelling decisions from other papers, as seen in Section 4.

4 EMS simulation model

This section covers the simulation model details, assumptions, verification and validation, and the package in which the model was implemented. At a minimum, an EMS simulation model needs to model the following objects: ambulances (representing both the ambulance and its assigned EMS staff), calls (representing both the emergency call and the patient), hospitals, and stations. A travel model is required in order to capture ambulance travel, as this plays a dominant role in the duration of ambulance service; for this work we chose to use a road network and allow some off-road travel when needed. Also, logic for dispatch and call queueing behaviour is required. Our model includes the option of dynamic ambulance redeployment, in which idle ambulances may be reassigned to different stations when prompted. Discrete event simulation is used, as it was a natural approach for this problem.

4.1 Emergency call generation

The location, priority, arrival time, and various durations pertaining to each emergency call need to be generated, and for simplicity, we assume that all of these generated values/parameters are independent of each other, e.g. location and arrival time are independent.

For generating emergency locations, a spatial probability density function is needed which indicates the relative likelihood of a call occurring at a given geographic location. The call generator reads in a raster file, which is a grid of rectangles where each rectangle holds a value, in this case the value is the relative call arrival rate. Given the number of calls to be generated, inverse transform sampling is used in order to efficiently generate many random locations simultaneously; for m calls and n grid rectangles, generating the locations has a run time of $\mathcal{O}(m+n)$.

The simulation package has three different emergency priorities: low, medium, and high priority (although this can easily be extended). Generating call priorities requires a categorical distribution (a discrete probability distribution where each outcome value has an associated meaning), which specifies the probability of each type of call priority. The probability that a patient will require transport to a hospital is generated from a Bernoulli distribution. Time related aspects of calls, such as inter-arrival time, ambulance dispatch delay, on-scene duration, and hospital handover duration, are also generated. Some of these durations that relate to medical service are actually dependent on the ambulance that is dispatched and the hospital that the patient is transported to, but for simplicity we set these durations to be dependent only on the emergency, though this can easily be changed. Any of the common continuous probability distribution functions can be used, such as uniform, normal, triangular, exponential, beta, Poisson, Erlang, Weibull, etc.

Calls are generated and saved to a file in a preparatory step before running the simulation. Alternatively, historical call data can used as input for the simulation.

4.2 Geographic coordinate system

Often, the location of objects in the simulation needs to be known, for instance when finding the nearest road to an emergency location so that a path can be determined. The simulation package uses latitude and longitude coordinates. The distances between two locations is calculated as $d = \sqrt{(\Delta\text{lat} \times c_{lat})^2 + (\Delta\text{lon} \times c_{lon})^2}$ where Δlat, Δlon are the differences in latitude and longitude of the two locations, and c_{lat}, c_{lon} are constants that scale the Δ values into distance units; these constants are required as input. This calculation provides a reasonably accurate distance value for regions that span less than a degree in latitude and longitude, are flat, and are not too close to either of the geographic poles (i.e. most cities).

4.3 Road network and path finding

A road network is used in order to model ambulance travel. The simulation package reads in a road network which has an underlying graph of nodes and arcs, where each node has coordinates. The graph underlying the road network must be strongly connected, in order to ensure that the ambulances can travel from any node to any other node. Arcs may have more than one associated travel time, in order to model different ambulance travel modes (ambulances may drive at regular traffic speed, or faster if using lights and sirens) and also model temporally varying travel times. The travel time along a path then depends on the travel mode and the travel start time.

For a given road network, all-pairs shortest paths (APSP) data is calculated and stored before the simulation begins so that simulation run time is reduced. Some road networks can be too large for the APSP data to be stored, so a road network with a reduced number of nodes is required. Often the underlying graph will have many intermediate nodes placed on road segments between intersections, and these intermediate nodes do not need to be considered when calculating the APSP. We follow a method by Henderson and Mason (Henderson and Mason 2004) in which a copy of the network is reduced down to just 'decision' nodes (nodes at which a driver has a choice of direction), and only calculate the APSP for this reduced network. Figure 2 shows an example of a full network and the resulting reduced network; note that the leaf nodes (nodes with only one adjacent arc) are also kept in the reduced network by the simulation, for easier reference between the two networks. Finding the shortest path from a start node to an end node can then be done by finding the nearest decision nodes from (to) the start (end) node, then deciding which of these decision nodes to travel through to give the shortest path. When possible, a path from the start node to the end node that does not include decision nodes also needs to be examined.

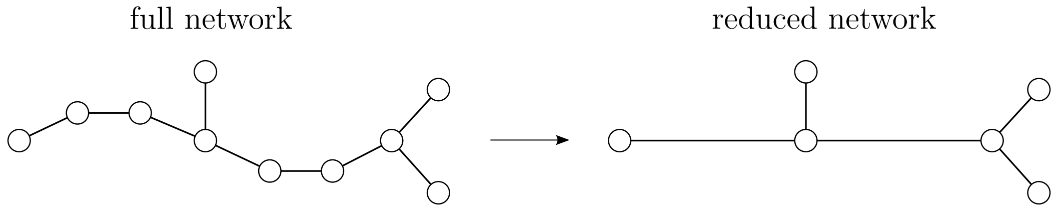

Figure 2: Example full network and corresponding reduced network

4.4 Simulation decision modules

Various modules are used for simulation logic for controlling the ambulance dispatching, call queueing, ambulance routing, and ambulance redeploying, as outlined in this subsection.

4.4.1 Dispatching

Whenever a call arrives, a decision needs to be made as to which ambulance should be dispatched to respond (unless the call is queued to be responded to later, because there are currently no available ambulances). In the current implementation the nearest available ambulance is dispatched. However, an ambulance that is travelling to a low priority call is considered for dispatching (in this case, diverting) if the new call has a higher priority than the call that the ambulance is currently responding to. An ambulance is also considered for dispatch to queued calls when the ambulance becomes available; if there is more than one queued call then they are serviced in order of emergency priority and then first-in-first-out order within each priority level.

4.4.2 Call queueing

If an emergency occurs at a time when there are no ambulances available to respond, the corresponding call is queued. A queue list is kept, for which calls are first sorted by priority (high to low), and then sorted by arrival time (earliest to most recent) within each priority level. When there are queued calls and an ambulance becomes available, it is dispatched to respond to the call at the front of the queue (earliest of the highest priority).

4.4.3 Routing

An ambulance travelling from one location to another requires a route, which consists of a path along the road network, and may include off-road travel to get on and off the road network. The reason for allowing off-road travel is that emergencies may occur away from the road network provided, and this travel is modelled for accuracy; this modelling consideration is more important for road networks that are less detailed. For travelling on the road network, it is assumed that the ambulance takes the fastest path available. For travelling on/off the network, ambulances may enter/leave the network at the nearest node to their start/end location, and the off-road travel speed is assumed to be constant for a given ambulance travel mode (regular, or lights and sirens). In order to prevent ambulances from travelling along a motorway and then directly off-road, nodes can be tagged as not being allowed to be used in order to gain off-road access. The destination of an ambulance, and thus its route, may be changed at any time.

4.4.4 Dynamic redeploying

Dynamic redeployment involves reassigning one or more idle ambulances to stations or standby locations, typically with the goal of improving performance, e.g. reducing response times. Dynamic redeployment models determine what the reassignment should be, usually by following a predetermined rule or solving an optimisation problem. Whenever an ambulance becomes busy or available, an event is triggered to consider redeploying that ambulance (if it is now available), or any other ambulances, depending on the redeployment model. For example, whenever an ambulance becomes busy or available, a compliance table redeployment model (also called system status management (Stout 1983)) looks at the new number of idle ambulances and gives a desired number of ambulances to allocate to each station, according to a precomputed table. For comparison, the Dynamic Maximum Expected Coverage Location Problem (DMEXCLP (Jagtenberg, Bhulai, and van der Mei 2015)) redeployment model considers redeployment only when an ambulance becomes available, and that ambulance is redeployed to the station for which it provides the largest improvement in the expected demand coverage. There are three dynamic redeployment models already implemented in the simulation package, a compliance table, a priority list (Zhang 2012) (this is a precomputed list that gives the priority in which stations should have ambulances redeployed to them), and an integer program by Zhang (Zhang 2012).

4.5 Assumptions and simplifications

The simulation model has some assumptions and simplifications, although most of the simplifications are simple to remove, we have not done so (yet) due to limited development time. For ambulances, we only model one type, which can respond to any emergency, and which operates 24 hours a day without breaks. Emergency calls are modelled as having parameters that are independent (e.g. location and arrival time of an emergency are independent), and these parameters do not depend on the ambulance or hospital that provide treatment (e.g. on-scene time is the same regardless of which ambulance was dispatched). Each emergency call requires exactly one ambulance to respond, and the call is not cancelled (a cancellation would mean that the ambulance would no longer need to respond). It is assumed that the direct off-road travel between a location and the nearest road network node is possible; this is not always correct, such as when the off-road travel crosses over water. The remaining assumptions and simplifications are more difficult to remove. For simplicity, travel times are modelled as being deterministic, rather than being stochastic. Also, the queueing of calls at the EMS call centre is not modelled; queueing only occurs in the dispatching of ambulances.

4.6 Verification and validation

In order to verify that the EMS operation, as in shown Figure 1, had been implemented correctly, an animation tool was added. The animation shows a map of the region, including the ambulances, hospitals, stations, and roads. The emergency locations appear as points on the map at their corresponding arrival time. From this animation, it is simple to verify that the ambulances follow the correct order of operations, and that all of the decision modules (dispatching, call queueing, routing, and dynamic redeploying) are working as intended.

To validate that the simulation model was correct, it was compared against a previously validated simulation model that had been implemented in the software BartSim (Henderson et al. 2000). Each simulation was given the same input data set of one month's worth of calls, and a detailed comparison between the output of each simulation showed that the ambulance dispatches were identical, and almost all of the emergency response times differed by less than one second. The emergency data set had 6544 calls; of these only 10 had response times differing by greater than one second between the two simulations. These 10 differences were due to minor variations in ambulance locations between the simulations at times of dispatch (we have not investigated the reason for this; it is possibly due to differences in numerical precision), leading to different routes being selected, and response time differences of up to 67 seconds.

4.7 Simulation package

Our simulation model has been implemented in the programming language Julia, chosen for its speed, ease of use (it is a high level programming language), and for straightforward mathematical optimisation through the JuMP modelling language that is embedded in Julia. JuMP provides a solver interface, allowing for problems that are linear, quadratic, non-linear, etc. to be solved.

The source code, along with instructions for installation and use of the simulation package, is available at https://github.com/samridler/AmbulanceSim.jl. In order to use the package, Julia will first need to be installed.

5 Simulation-based optimisation example

Simulation can be used not just for evaluating performance of a system, but also for use in a simulation-optimisation framework, in order to optimally (within some level of certainty) or heuristically improve system performance.

One of the long-term decisions required to be made for EMS operations is the deployment of ambulances to stations. A deployment policy allocates each ambulance to a station, which the ambulance may return to when not busy. For a realistic sized problem, the number of ambulances and stations leads to a large number of possible deployment policies, e.g. with n homogeneous ambulances and m stations with unlimited ambulance holding capacity, there are $\binom{n+m-1}{m-1}$ unique deployment policies. As a result, in order to optimise a deployment policy, either an analytic solution to an approximation of the system is needed, or a method of partially searching through solution space with a more detailed system model such as a simulation model.

This section demonstrates how the EMS simulation package may be used in a simulation-based optimisation framework, using the example of selecting the “best” of many randomly generated ambulance deployment policies. Here, the objective is to select the deployment policy that minimises the average call response time. The remainder of this section details the experiment and results for one region.

5.1 Experiment model

For the experiment, a simulation of EMS operations in Auckland, New Zealand was built. The region considered has approximately 1.42 million people, and was modelled with parameters that mimicked those used for BartSim (which was built to model Auckland). The model included 3 hospitals, 17 stations, and 17 ambulances. The emergency call inter-arrival time was modelled with an exponential distribution for an arrival rate of 300 calls per day. Each call was assigned a priority of high, medium, or low, with probability 57%, 24.5%, and 18.5%, respectively. The probability of a call occurring in a geographic location was modelled as being proportional to the population density (data from koordinates.com) at that location. Dispatch delay was modelled with a normal distribution with a mean of 2 minutes and a standard deviation of 1 minute, truncated to be between 20 seconds and 5 minutes. On-site treatment time (and also the handover time at the hospital) was modelled with an exponential distribution with a mean of 12 minutes, truncated to be between 2 minutes and 30 minutes. The probability that the patient requires transport to a hospital was set to 80%. The road network (from openstreetmap.org) had 62 637 nodes and 106 400 arcs. The corresponding reduced network had 4 695 nodes and 10 237 arcs. Travel times were modelled as being deterministic and temporally static. The lights and sirens speed (for responding to high priority emergencies) was set to be 33% faster than regular travel speed.

5.2 Experiment

For the experiment, 100 deployment policies are randomly generated. We then simulate each using a single set of calls and then save the resulting call response times for each policy. Within each deployment policy, each ambulance is randomly allocated to one of the stations. The objective function chosen was to minimise the average emergency response time. In practice, this is not always a good objective function to use, as it will heavily favour the placement of ambulances in areas of high demand, at the expense of providing poor service for lower demand areas (e.g. rural zones). Regardless, we use this objective function for the experiment, in order to avoid selecting a policy that leave patients waiting a long time (on average) for ambulances.

In order to estimate the expected response time from employing each of the 100 deployment policies, many independent estimates of the average call response time for each policy are needed. Rather than running many short simulation replications (for each policy) and removing the initial transient period from every replication, we use the batch means method (Alexopoulos and Seila 1996). The batch means method involves running one long simulation run and only removing the initial transient once, then splitting observations into batches and calculating the mean value for each batch, thus reducing the computation required. The batches each need to include enough observations to allow the mean values from the batches to be treated as independent and identically distributed random variables with a mean equal to the true mean.

For the simulation experiments, 150 000 generated emergencies (approximately 500 day's worth) were used, and the emergency response times were recorded for each policy. To determine the duration of the transient period, a graphical procedure by (Welch 1981), (Welch 1983) as presented by (Law 2007) was followed, resulting in a conservative estimate of the transient period being one day. This transient period was removed, and call response times between the end of the transient period and the last call were then batched by week. To check that the batches were large enough to give effectively independent estimates of the true mean response time, serial autocorrelation of the batch means was tested by fitting an AR(0) model (i.e. constant mean with white noise), and using the Durbin-Watson test to detect serial autocorrelation in the residuals of the model. For the 100 deployment policies, only 5 had p-values less than 0.1 (minimum p-value was 0.038), indicating that the week-long batches were large enough to not have significant serial autocorrelation.

5.3 Results

Figure 3 shows the distribution of batch mean response times from employing each of the randomly generated deployment policies, in order of increasing average call response times.

Of the 100 deployment policies tested, the mean response time ranged from 7.70 minutes (best policy) to 12.60 minutes (worst policy). The best policy placed most of the ambulances in urban areas, while the worst policy placed most of the ambulances in rural areas, with one station receiving 6 of the 17 ambulances.

The same deployment policies were also simulated with a different set of 300 000 calls, resulting in an almost identical ranking of policies when ordered by average response time (Spearman's rank order coefficient of 0.9996), indicating that the first

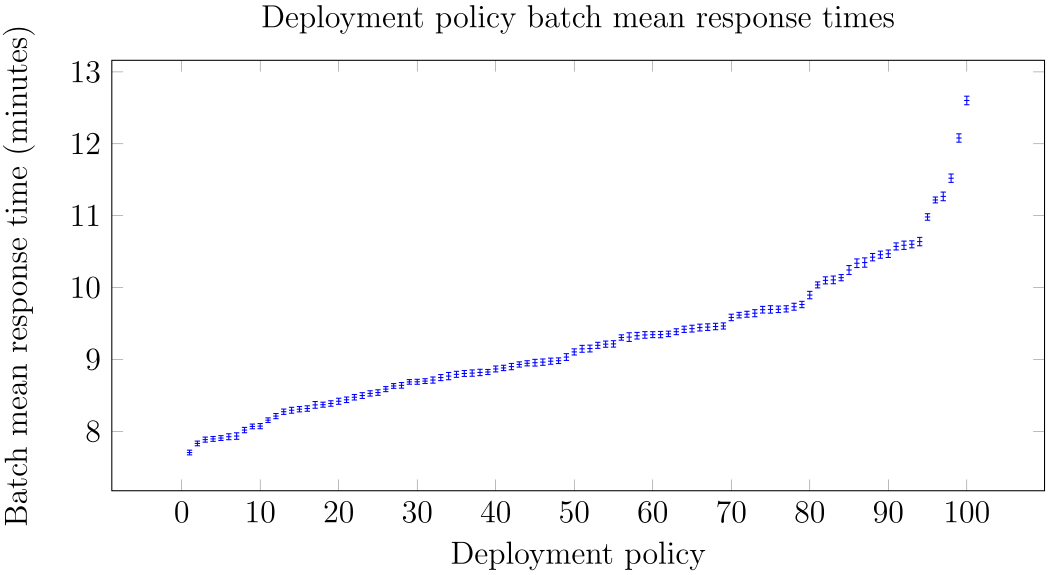

Figure 3: Batch mean response times for 100 randomly generated deployment policies; for each policy, the mean ±2 times the standard error of the mean is shown

call set with 150 000 emergencies was sufficient in length in order to avoid significant training bias.

While the example of randomly generating deployment policies and selecting the best of those generated is not recommended in order to create an optimised deployment policy, it does show some of the steps involved in using simulation in an optimisation framework. A better approach for this problem may be to use a local search heuristic as done by Zhang (Zhang 2012), in which an initial ambulance deployment policy is given and the neighbourhood search allows an ambulance from any one station to be moved to any other station, with changes to the policy being kept when they improve estimated performance.

6 Conclusions

We have developed a simulation package for EMS systems in the programming language Julia. The simulation model is detailed, which draws from common features of other EMS models in the literature; many of the simplifications in the model are relatively simple to remove. To show the use of the package, an example of its use in an optimisation framework was given. We hope that the simulation package proves useful to other researchers in this field.

References

Aboueljinane, Lina, Evren Sahin, and Zied Jemai. 2013. “A review on simulation models applied to emergency medical service operations.” *Computers & Industrial Engineering* 66 (4): 734–750.

Alexopoulos, Christos, and Andrew F Seila. 1996. “Implementing the batch means method in simulation experiments.” *Proceedings of the 28th conference on Winter simulation.* IEEE Computer Society, 214–221.

Andersson, T, and Peter Värbrand. 2007. “Decision support tools for ambulance dispatch and relocation.” *Journal of the Operational Research Society* 58 (2): 195–201.

Aringhieri, Roberto. 2010. “An integrated DE and AB simulation model for EMS management.” *Health Care Management (WHCM), 2010 IEEE Workshop on.* IEEE, 1–6.

Carter, Grace M, Jan M Chaiken, and Edward Ignall. 1972. “Response areas for two emergency units.” *Operations Research* 20 (3): 571–594.

Daskin, Mark S. 1983. “A maximum expected covering location model: formulation, properties and heuristic solution.” *Transportation Science* 17 (1): 48–70.

Henderson, Shane G, and Andrew J Mason. 2004. “Ambulance service planning: simulation and data visualisation.” In *Operations Research and Health Care*, 77–102. Springer.

Henderson, Shane G, Andrew J Mason, et al. 2000. “BartSim: A tool for Analysing and Improving Ambulance Performance in Auckland, New Zealand.” *Proc. of the 35th Annual Conference of the Operational Research Society of New Zealand, Wellington, New Zealand.* 57–64.

Jagtenberg, CJ, S Bhulai, and RD van der Mei. 2015. “An efficient heuristic for real-time ambulance redeployment.” *Operations Research for Health Care* 4:27–35.

———. 2016. “Dynamic ambulance dispatching: is the closest-idle policy always optimal?” *Health Care Management Science*, pp. 1–15.

Kergosien, Yannick, Valérie Bélanger, Patrick Soriano, M Gendreau, and A Ruiz. 2015. “A generic and flexible simulation-based analysis tool for EMS management.” *International Journal of Production Research*, pp. 1–18.

Larson, Richard C, and Amedeo R Odoni. 1981. *Urban operations research.*

Law, Averill M. 2007. *Simulation modeling and analysis.* 4. McGraw-Hill.

Mason, Andrew James. 2013. “Simulation and real-time optimised relocation for improving ambulance operations.” In *Handbook of Healthcare Operations Management*, 289–317. Springer.

Pinto, L.R., P.M.S. Silva, and T.P. Young. 2015. “A generic method to develop simulation models for ambulance systems.” *Simulation Modelling Practice and Theory* 51:170–183. cited By 2.

Stout, Jack L. 1983. “System status management.” *Journal of Emergency Medical Services* 8 (5): 22–3.

Welch, Peter D. 1981. “On the problem of the initial transient in steady-state simulation.” *IBM Watson Research Center.*

———. 1983. “The statistical analysis of simulation results.” *Computer Performance Modeling Handbook* 22:268–328.

Zhang, Lei. 2012. “Simulation optimisation and Markov models for dynamic ambulance redeployment.” Ph.D. diss., Department of Engineering Science, University of Auckland.