This article was published in an Elsevier journal. The attached copy is furnished to the author for non-commercial research and education use, including for instruction at the author’s institution, sharing with colleagues and providing to institution administration.

Other uses, including reproduction and distribution, or selling or licensing copies, or posting to personal, institutional or third party websites are prohibited.

In most cases authors are permitted to post their version of the article (e.g. in Word or Tex form) to their personal website or institutional repository. Authors requiring further information regarding Elsevier’s archiving and manuscript policies are encouraged to visit:

http://www.elsevier.com/copyright
A multiple dispatch and partial backup hypercube queuing model to analyze emergency medical systems on highways

Ana Paula Iannoni, Reinaldo Morabito *

Department of Production Engineering, Federal University of Sao Carlos, SP, Brazil

Received 30 December 2005; received in revised form 10 April 2006; accepted 23 May 2006

Abstract

The hypercube is a spatially distributed queuing model based on Markovian analysis approximations, used to analyze the configuration and operation of server-to-customer emergency systems. In the present study we adapted the model to analyze emergency medical systems (EMS) on highways, which operate within particular dispatching policies. The study takes into consideration that: the emergency calls are of different types; the servers are distinct (e.g., rescue ambulances, medical vehicles); only certain servers in the system can service calls in a given region (partial backup); and, depending on the type of call, one or more identical or distinct servers are immediately dispatched to service such calls (multiple dispatch). We also consider that the arriving calls take place either along the highway or at the home location of a server – in which case the server does not need to travel to the call location. Finally, we analyzed the computational results of applying such an approach to the case study of an EMS operating on Brazilian highways.

© 2007 Elsevier Ltd. All rights reserved.

Keywords: Emergency medical systems; Hypercube queuing model; Ambulance location; Highways

1. Introduction

In emergency medical systems (EMS) the delays in the response times are of major concern since they might mean the difference between life and death of individuals involved in accidents. When designing or modifying the configuration of EMS, managers try to balance investments and benefits of improving the operation of these systems. Since EMS are server-to-customer systems, the analysis of their operations requires that probabilistic factors related to the spatial (location) and temporal (time) distribution of both user calls and system servers, be modeled. In the literature, most models that analyze emergency systems take into account only randomness in the server availability, not considering other important probabilistic variables of the system.
Studies by Swersey (1994), Owen and Daskin (1998) and Brotcorne et al. (2003) present revisions of the classic location models for the analysis of the emergency systems developed during the last few decades.

The hypercube model (Larson, 1974, 1975; Larson and Odoni, 1981), which is based on the spatially distributed queuing theory and Markovian analysis approximations, has been one of the most effective descriptive approaches to analyze emergency response systems. This model enables the representation of the EMS uncertainties, besides considering the identity of the servers and the cooperation among them. The basic idea is to expand the state-space description of a multi-server queuing system (e.g., uncertainties, besides considering the identity of the servers and the cooperation among them. The basic idea is to expand the state-space description of a multi-server queuing system (e.g., uncertainties, besides considering the identity of the servers and the cooperation among them. The basic idea is to expand the state-space description of a multi-server queuing system (e.g., $M/M/N/x$ or $M/M/N/N$ where $N$ is the number of servers) in order to represent each server individually and to incorporate the complex dispatching policies involved. The model requires the solution of linear systems of $O(2^N)$ equations, where the variables involved are the equilibrium state probabilities of the system. With these probabilities, it is possible to calculate different critical performance measures, such as mean user-response times, server workloads, number of dispatches per server in each region, among other measures.

Several studies, such as Larson (1975), Halpern (1977), Chelst and Barlach (1981), Larson and Mcknew (1982), Jarvis (1985), Burwell et al. (1993) and Mendonça and Morabito (2001), have extended the original hypercube model to relax some of its limiting assumptions or to improve its computational efficiency in analyzing police and ambulance emergency response systems. In particular, Chelst and Barlach (1981) extended the hypercube model to consider the simultaneous dispatch of two identical police patrols, and Mendonça and Morabito (2001) modified the model considering the single dispatch of identical ambulances and partial backup in EMS on highways. Other studies have been successful in combining the hypercube model with optimization procedures, such as Battà et al. (1989), Saydam and Aytug (2003), Chiyoshi et al. (2003) and Galvão et al. (2005). Examples of the hypercube model applications in urban EMS in the United States can be found in Larson and Odoni (1981), Chelst and Barlach (1981), Brandeau and Larson (1986), Burwell et al. (1993) and Sacks and Grief (1994). More recently, the hypercube has been studied as a deployment model in response to terrorist attacks and other major emergencies (Larson, 2004). In Brazil, the hypercube model has been applied to analyze urban EMS (e.g., Takeda et al., 2007) and EMS on highways (Mendonça and Morabito, 2001; Iannoni, 2005).

In the present study we modify the hypercube model to analyze EMS on Brazilian highways, considering complex dispatching policies. In the first model, we extend the multiple dispatch hypercube model proposed by Chelst and Barlach (1981) to analyze EMS on highways with different types of calls and servers, and a particular operations policy involving partial backup (because of limitations of travel distance or time), zero-line capacity and single and multiple dispatching of either identical or distinct servers (e.g., rescue ambulances, medical vehicles) depending on the type of call. This model is then extended to incorporate a third status of servers (besides the usual two: idle or busy), to specify when they provide medical services to emergency requests at their home bases – as this type of service does not involve a server’s trip to the call location and, in general, it is not related to accidents along the highway. After applying the two models to the case study of an EMS operating on Brazilian highways, we analyze the computational results and show that the second model represents this EMS more accurately.

This paper is organized as follows: in Section 2 we describe the EMS under consideration; in Section 3 we discuss the extensions of the hypercube model to this EMS; and in Section 4 we analyze some results obtained to evaluate the main performance measures of this system, using sample analysis to validate the models. Finally, in Section 5 we present concluding remarks and perspectives for future studies.

2. The EMS under consideration

EMS on highways are typically zero-line capacity systems, since if a call arrives in the system when its candidate ambulances are busy, it is immediately transferred to another system, such as a local hospital or a police station, which is usually unable to provide the same service quality. In general, an EMS’ ambulance provides the first medical treatments to the individuals involved in the accident, transports them to the nearest hospital (if necessary), and then goes back to its home base on the highway. Furthermore, the ambulance can also provide medical services to users that stop at its base asking for help. The operations center, usually located on the highway or in a strategic city, handles the calls, dispatches the ambulances and tracks the movements of the ambulances.
The operation of many EMS on Brazilian highways is under the management of private organizations as part of privatization contracts with the Brazilian government. Private operators are responsible for all aspects of running an EMS, including system design, maintenance, operations planning and response to user’s emergency calls on the highways. In Brazil, the number of accidents on highways is still of major concern, especially on highways managed by non-private organizations. In this paper, we study an EMS operated by a private firm, and based on certain stretches of highways located in the state of Sao Paulo.

This EMS has six servers located, comprising five identical rescue ambulances and a vehicle called the medical vehicle. Such vehicle transports specialized medical personnel (doctors, nurses, rescuers), some basic medications and tools for pre-clinical care. This vehicle does not transport the individuals involved in accidents. Conversely, the rescue ambulance transports the patients injured in the accidents, rescuers, clinical and pre-hospital care equipment and other heavier equipment (e.g., hardware breakers, fire control equipment, etc.). Each of these ambulances is located in one of the five fixed bases along the highways; the medical vehicle is located at Base 1, sharing this home location with ambulance 1. The operations center is also located in one of the bases. Fig. 1 illustrates the configuration of this EMS on the stretches of highways in Sao Paulo state.

The dispatching of different types of servers as a distinguishing feature for the operation of some highways’ EMS can also be found in some Brazilian urban EMS. In these EMS, the ambulances can be either advanced support vehicles (ASV) or basic support vehicles (BSV), the calls are divided into two classes: advanced calls (preferentially serviced by an ASV) and basic calls (preferentially serviced by a BSV) (see, e.g., Takeda et al., 2007). On some of the busiest highways, the EMS can also use helicopters to rescue people involved in accidents.

Given the characteristics of the presently studied EMS servers (rescue ambulances and medical vehicle), and the limitations of travel distance and time, the dispatching of different servers depends on the nature of the emergency call and its location along the highway. As described by the managers and operators of this system, the dispatching policy is given as follows:

(i) Single dispatching of either an ambulance or the medical vehicle. Ambulances respond to calls requiring rescue and transport of patients and/or specialized equipment. Certain kinds of calls require single dispatch: when they are received at the operations center, the nearest available ambulance is immediately dispatched to the call location. When the nearest ambulance is busy, the second nearest ambulance (called backup) is dispatched. If this second ambulance is also busy, the call is transferred to another emergency system (even if there are other ambulances available) and it is considered lost to the system.

When the arriving call is of a nature that requires solely the services of a doctor on-scene to manage the necessary medical procedures, and it does not require specialized equipment and/or transport of patients, the medical vehicle is preferentially dispatched to service the call, depending on the location. In this case, when the medical vehicle is busy, the nearest ambulance (backup) is dispatched, if available. Otherwise, if the backup is also busy, the call is considered lost.

Fig. 1. EMS on stretches of highways in Sao Paulo state.
(ii) Double dispatching – that of two ambulances or dispatching of the medical vehicle and one ambulance. When an emergency call that requires a double dispatch arrives, the two nearest ambulances are immediately dispatched to the call location. If only one of them is available then, depending on the location of the call, either the available ambulance is assigned as a single dispatch (possibly with the assistance of another emergency system), or the third nearest ambulance (backup) is dispatched. If the two closest ambulances are busy, depending on the call location, either the call is considered lost by the system or the third nearest ambulance is dispatched. If this ambulance is also busy, the call is lost. When the call requires the services of a doctor on-scene and other services that can be only provided by an ambulance (e.g., rescue and transport of patients), the medical vehicle and the nearest ambulance are preferentially dispatched, depending on the call location. In this case, if one of them is busy, the second nearest ambulance (backup) is dispatched, if it is available. When it is not possible to dispatch the medical vehicle, the ambulances (staffed with rescuers and medical equipment) are sent, and the doctor follows the medical procedures by a radio system.

(iii) Triple dispatching – that of three ambulances, or the medical vehicle and two ambulances. When an emergency call requires the assistance of three (or more) servers (e.g., according to the number of individuals involved in the accident and the type of accident), either the three nearest ambulances or the medical and the two nearest ambulances are immediately dispatched, depending on the call location. If only two of them are available, they are assigned as a double dispatch (i.e., possibly with the assistance of another emergency system, since the fourth nearest server is never dispatched). When only one of them is available, it is dispatched as a single dispatch. If the three closest ambulances, or the medical vehicle and the two nearest ambulances are busy, the call is considered lost. When the call requires the services of a doctor on-scene, the medical vehicle and the two nearest rescue ambulances are preferentially dispatched, depending on the call location. In this case, if one of them is busy, the third nearest ambulance (i.e., the fourth server – backup) is not dispatched. Similarly to the double dispatching, when the three nearest ambulances are preferentially dispatched, and in the case it is not possible to dispatch the medical vehicle (e.g., due to the distance of the call location to the medical vehicle’s base, or when the medical vehicle is busy), the doctor coordinates the medical procedures carried out by the ambulance personnel via a radio system.

In addition, the servers of the system can be busy servicing a call without leaving its home base. For example, when users stop at the ambulance’s base asking for help (e.g., epileptics, heart attacks, alcoholics, etc.). As in such events the server does not need to travel to the call location (i.e., there is no travel time for walk-in callers), the mean service time (which, in general, includes the set-up time, the travel time to the call location, the on-scene service time and the travel time back to the base) is distinctive for this type of calls. However, if this call arrives in a base and the base’s ambulance is already busy servicing an accident, the call is transferred to another emergency system and it is considered lost to the system (i.e., there is no backup for this call).

This system can be conceptualized as a spatially distributed multi-server loss (zero-line) queuing system, with calls and serves of different types and a specific dispatching policy, which considers that only some servers can travel to certain regions (partial backup), and a call may be considered lost even if there are other servers available. Therefore, this dispatching policy violates some assumptions of the basic hypercube model in Larson (1974). Such assumptions are that: any server would travel (respond) to any call location; only one server would be dispatched to a call; any call could be serviced by a backup server (whenever the preferential server is busy), all servers are identical and each server has only two possible states: idle or busy (i.e., servicing a call outside its base). In the next section, we discuss how the hypercube model can be adapted and applied to analyze EMS on highways like the one of the present study.

3. The adapted hypercube model

The name hypercube derives from the state space of the system. For the zero-line capacity system, it can be described as the vertices of the N-dimensional unit hypercube in the positive orthant, and each vertex corresponds to a particular combination of servers’ status (Larson, 1974). In the basic model, at any given time instant, the status of each server is either idle (0) or busy (1), and for N servers, there are $2^N$ possible states.
(vertices) for the system. For example, the state \( \{101\} \) corresponds to a three-server system, with servers 1 and 3 busy and server 2 idle, and the state space is given by the vertices of a cube. If the system has more than three servers, we have a hypercube. The main assumptions of the multiple dispatch and partial backup hypercube model for application on highways are:

- The highways are divided into \( N_A \) geographical atoms, which correspond to independent sources of calls. At each atom \( j \), calls arrive according to a Poisson process, independently from the other atoms. The calls are classified according to the type of service and dispatch (single, double or triple dispatches for calls along the highway, and no dispatches for requests at the server bases).
- There are \( N \) servers spatially distributed along the highways and, when idle, they wait for calls in their home bases. The servers are of different types (rescue ambulances and medical vehicle) and each server can only be dispatched to their predetermined primary and backup atoms. A server’s preferential area consists of those atoms to which such a server would be dispatched, when available, even if all other servers were also available.
- For each atom, there is a dispatch preference list ranking the servers to be dispatched, depending on the call type. The preferential (usually the nearest) server of a given atom corresponds to the first on that list. Given the limitations of travel distance or time, only some servers can travel to certain regions (partial backup), and a call may be considered lost even if there are other servers available.
- The service time of each server includes the set-up time, the travel time to the call location, the on-scene service time and the travel time back to its base. In general, each server \( i \) has a distinct mean service time, \( 1/\mu_i \). The model assumes negative exponential service times, although reasonable deviations from this assumption have been found not to greatly alter the accuracy of the model.
- The travel time between two regions is known or it can be estimated by geometric probability, which corresponds to a collection of special geometrically oriented tools, methods, and (sometimes) tricks that facilitate the analysis when the standard techniques of probabilistic analysis are cumbersome in a geometric setting (Larson and Odoni, 1981). The service time variations that are due to variations in the travel times to the call locations are considered to be of second order, when compared to variations resulting from the sum of times spent on set-up, on-scene service, and on travel back to the bases. It is important to note that, given that the system does not allow queuing, the mean user-response time is simply composed of the mean set-up time and the mean travel time to the call locations.

3.1. Model 1 – multiple dispatch and partial backup hypercube model

In the first model (model 1), the calls in each atom are classified in terms of the number of servers (single, double or triple dispatch) and the type of servers (rescue ambulances and the medical vehicle) required. The dispatching policy is complex since the types of calls may need: one server, two identical servers (i.e., two rescue ambulances), two distinct servers (one rescue ambulance and the medical vehicle), three rescue ambulances, and two rescue ambulances and the medical vehicle. Moreover, the partial backup policy increases this complexity provided that there can be calls with only two servers in its dispatching preference list (partial backup) and calls with three servers (e.g., when the closest base to the accident is the home station of two distinct servers).

Given these particularities of the operations policy and different types of servers and calls in each atom \( j \) of the EMS, the model assumes six types of calls:

(i) Type 1a calls (with arrival rate \( \lambda_j^{[1a]} \)) require the single dispatch of an ambulance, and there are two candidate servers in its dispatching preference list (the nearest and the backup ambulances). For a type 1a call, the highest ranked ambulance is dispatched; when this ambulance is busy, the next one from the list is dispatched (backup ambulance). If the backup is also busy, the call is lost (partial backup).

(ii) Type 1b calls (with arrival rate \( \lambda_j^{[1b]} \)) require the single dispatch of the medical vehicle, and there are two candidate servers in its dispatching preference list (the medical vehicle and the backup ambulance). Similarly to type 1a calls, when the medical vehicle is busy, the backup ambulance is dispatched (and the doctor instructs the ambulance personnel by radio). If the backup is also busy, the call is lost.
(iii) Type 2a calls require the double dispatch, i.e., that of two identical ambulances. Depending on the location of the call, there are two sub-cases, called type 2a’ and 2a” calls (with arrival rates \( \lambda_{2a'} \) and \( \lambda_{2a''} \), respectively). In the first sub-case (type 2a’), there are two candidate servers (the two nearest ambulances) in the dispatching preference list. These two ambulances are simultaneously dispatched; however, when only one of the ambulances is available, a single dispatch is assigned to the call. If the two ambulances are busy, the call is lost, since a third ambulance is not dispatched. In the second sub-case (type 2a’’), there are three candidate servers (the two nearest ambulances and the backup one) in the dispatching preference list. The backup ambulance is sent whenever any of the two first in the list is busy. Moreover, if both of them are busy, the backup is sent as a single dispatch. If the backup is also busy, the call is lost.

(iv) Type 2b calls (with arrival rate \( \lambda_{2b} \)) require the double dispatch, including the medical vehicle and an ambulance. In the dispatching preference list for these calls there are three candidate servers (the medical vehicle and the nearest and the backup ambulances). Similarly to type 2a” calls, the backup ambulance is sent if any of the two first vehicles in the list is busy. If both of those vehicles are busy, the backup is sent as a single dispatch. Whenever the backup vehicle is also busy, the call is lost.

(v) Type 3a calls (with arrival rate \( \lambda_{3a} \)) require the triple dispatch, involving three identical ambulances. There are three candidate servers in its dispatching preference list: the three nearest ambulances, which are simultaneously dispatched. However, when only one or two of them are available, a double or even a single dispatch may be assigned to such calls. If the three ambulances are busy, the call is lost, since a fourth ambulance is not to be dispatched.

(vi) Type 3b calls (with arrival rate \( \lambda_{3b} \)) require the triple dispatch, which includes the medical vehicle and two ambulances. There are three candidate servers (the medical vehicle and the two nearest ambulances) in its dispatching preference list. Similarly to type 3a calls, these three ambulances are simultaneously dispatched and, when only one or two of them are available, these calls are assigned a single or double dispatch. If the three ambulances are busy, the call is lost.

In the present system, an ambulance can be a backup to the medical vehicle in call types 1b and 2b. Note that this simplifies the analysis. If the system were different, we would have to consider a list of backup servers for the ambulances and the medical vehicle, individually. The total arrival rate at atom \( j \) is defined as \( \lambda_j = \lambda_j^{[1]} + \lambda_j^{[2]} + \lambda_j^{[3]} \), where \( \lambda_j^{[1]} = \lambda_j^{[1a]} + \lambda_j^{[1b]} \), \( \lambda_j^{[2]} = \lambda_j^{[2a']} + \lambda_j^{[2a'']} + \lambda_j^{[2b]} \) and \( \lambda_j^{[3]} = \lambda_j^{[3a]} + \lambda_j^{[3b]} \) are, respectively, the arrival rates of calls requiring single (type 1 calls), double (type 2 calls) and triple (type 3 calls) dispatches at atom \( j \).

Regarding the service process, type 1 calls are serviced by a single server \( i \) with mean service rate \( \mu_i \). Type 2 calls are serviced by two servers \( j \) and \( k \), which operate independently with mean service rates \( \mu_j \) and \( \mu_k \), respectively. Therefore, two servers servicing a type 2 call can be treated the same as two servers servicing two separated type 1 calls. As discussed by Chelst and Barlach (1981), if the service of calls requiring double dispatch were modeled differently, we would have to consider an additional state for each server, in order to differentiate servers busy by type 1 and type 2 calls – in which case, the state space of the system would increase to \( 3^N \). Similarly, type 3 calls are serviced by three servers \( i \), \( k \), and \( l \), which operate independently, with mean service rates \( \mu_i \), \( \mu_k \) and \( \mu_l \), respectively.

For convenience, we use an illustrative simple example to describe how this model can be applied to analyze EMS on highways (such as the EMS of the present study). This example, depicted in Fig. 2 has \( N_A = 3 \) atoms

![Fig. 2. EMS on highway – multiple dispatch and partial backup with \( N_A = 3 \) atoms and \( N = 4 \) distinct servers.](image-url)
and $N = 4$ servers: a medical vehicle (server 1, denoted in the figure as MV) and three rescue ambulances (servers 2, 3 and 4, denoted in the figure as R1, R2 and R3). The medical vehicle and the first ambulance are located in the same base at atom 1; the second ambulance is located at atom 2’s base; and the third ambulance, at atom 3’s base. In each atom $j$, calls can be of types 1 (1a or 1b), 2 (2a, 2a’ or 2b) and 3 (3a or 3b), with corresponding arrival rates $\lambda^{1a}_j$, $\lambda^{1b}_j$, $\lambda^{2a}_j$, $\lambda^{2a’}_j$, $\lambda^{2b}_j$, $\lambda^{3a}_j$ and $\lambda^{3b}_j$.

In order to model the dispatch of different types of servers to the same emergency call, we introduced separate dispatching preference lists to each type of call and subdivided each atom according to the number of lists. This procedure is referred to as layering (Larson and Odoni, 1981; Takeda et al., 2007). For the system example, each atom $j$ may be divided into two layers: layer $a$ (sub-atom $ja$) for type 1a, 2a’, 2a” and 3a calls serviced only by rescue ambulances; and layer $b$ (sub-atom $jb$) for type 1b, 2b and 3b calls serviced by the medical vehicle and the rescue ambulances. In other words, without loss of generality, each atom $j$ is considered as two sub-atoms ($ja$ and $jb$), while the original system with three atoms is analyzed as a system with six sub-atoms.

Table 1 shows the dispatching preference list for each of the six sub-atoms. To facilitate presentation, we do not consider all types of calls in each sub-atom (note, for example, that there are no calls of type 3a arriving at sub-atom 3a). For instance, take sub-atom 1a, where a type 1a call is preferentially served by server 2 (single dispatch of ambulance R1) and its backup is server 3 (ambulance R2), while a type 2a call is served by servers 2 and 3 (double dispatch of ambulances R1 and R2), having no backup. Similarly, take sub-atom 1b, where a type 1b call is preferentially served by server 1 (single dispatch of the MV) with its backup being server 2 (ambulance R1), while a type 2b is preferentially served by servers 1 and 2 (double dispatch of the MV and ambulance R1), with its backup being server 3 (ambulance R2); in sub-atom 1b, a type 3b call is served by servers 1–3 (triple dispatch of the MV and ambulances R1 and R2), but has no backup. Note that for sub-atom 2a, a type 2a” call is preferentially served by servers 3 and 2 (double dispatch of ambulances R2 and R1), having the server 4 (ambulance R3) as backup, while for sub-atom 3a, a type 1a call is preferentially served by server 4 (ambulance R3) whose backup is solely server 3 (ambulance R2) – which means that server 2 (ambulance R1) in the last column of the table refers only to type 3a calls at sub-atom 3a.

### 3.1.1. Equilibrium probability equations of model 1

Applying Markovian analysis and assuming that the system attains steady state, the equilibrium probability equation for each state is formulated taking into account that the inflow rate (the probability of the system being in another state, times the transition rate from that state to the current state) is equal to the outflow rate (the probability of the system being in the current state, times the transition rate from this to another state). Below, we illustrate how the equilibrium equations can be formulated by analyzing one of the $2^4 = 16$ possible states of the system example (the analysis of the remaining states is similar). For example, the equilibrium equation of state $\{1110\}$ is given by the equation below:

$$
\begin{align}
&\left(\lambda^{1a}_{3a} + \lambda^{2a’}_{2a} + \lambda^{3a}_{2a} + \lambda^{2a’}_{2a} + \mu_1 + \mu_2 + \mu_3\right) \cdot P_{\{1110\}} \\
&= \left(\lambda^{1b}_{1b} + \lambda^{3b}_{2b}\right) \cdot P_{\{0000\}} + \left(\lambda^{1a}_{1a} + \lambda^{2a’}_{1b} + \lambda^{3b}_{2b} + \lambda^{2a’}_{2a} + \lambda^{2a”}_{2a} + \lambda^{2b}_{2b} + \lambda^{3b}_{2b}\right) \cdot P_{\{0100\}} + \left(\lambda^{1a}_{1a} + \lambda^{2a’}_{1b} + \lambda^{3b}_{2b} + \lambda^{2a”}_{2a} + \lambda^{2b}_{2b} + \lambda^{3b}_{2b}\right) \cdot P_{\{0101\}} \\
&+ \left(\lambda^{1a}_{1a} + \lambda^{2a’}_{1b} + \lambda^{3b}_{2b}\right) \cdot P_{\{0110\}} + \left(\lambda^{1a}_{1a} + \lambda^{3b}_{2b}\right) \cdot P_{\{0111\}} + \left(\lambda^{1a}_{1a} + \lambda^{2a’}_{1b} + \lambda^{3b}_{2b}\right) \cdot P_{\{1000\}} + \left(\lambda^{1a}_{1a} + \lambda^{2a’}_{1b} + \lambda^{3b}_{2b}\right) \cdot P_{\{1001\}} + \left(\lambda^{1a}_{1a} + \lambda^{2a’}_{1b} + \lambda^{3b}_{2b}\right) \cdot P_{\{1010\}} + \left(\lambda^{1a}_{1a} + \lambda^{3b}_{2b}\right) \cdot P_{\{1011\}} + \mu_4 \cdot P_{\{1111\}}.
\end{align}
$$

(1)
Note on the left side of this equation (outflow rate of state $\{1110\}$) that only type $2a'$ calls generated at sub-atom $2a$ and calls generated at sub-atoms $3a$ and $3b$, can be serviced in the system when it is at state $\{1110\}$, since only server $4$ is available (see Table 1). Conversely, we verified on the right side of the equation (inflow rate of state $\{1110\}$) that the transitions into state $\{1110\}$ are

- $\{0000\} \rightarrow \{1110\}$: servers $1$–$3$ go simultaneously into service when a type $3b$ call arrives at sub-atom $1b$ or $2b$.
- $\{1000\} \rightarrow \{1110\}$: servers $2$ and $3$ go simultaneously into service when a type $2a'$ call arrives at sub-atom $1a$ or $2a$, or a type $2a'$ call arrives at sub-atom $2a$, or a type $2b$ or $3b$ call arrives at sub-atom $1b$ or $2b$. Note that type $3$ calls are serviced by the two ambulances available in its dispatching list.
- $\{0100\} \rightarrow \{1110\}$ and $\{0010\} \rightarrow \{1110\}$: servers $1$ and $3$, or servers $1$ and $2$, go simultaneously into service when a type $2b$ or $3b$ call arrives at sub-atom $1b$ or $2b$, respectively, and one of its two preferential servers is busy (the third nearest server is also dispatched).
- $\{1010\} \rightarrow \{1110\}$: server $2$ goes into service when a type $1a$ or $2a'$ call arrives at sub-atom $1a$ or $2a$, a type $2b$ or $3b$ call arrives at sub-atom $1b$ or $2b$, or a type $1b$ arrives at sub-atom $1b$ (since server $2$ is the backup ambulance). Note that, from their dispatching list, types $2a'$, $2b$ and $3b$ calls find only one server available and they are served by a single dispatch.
- $\{1100\} \rightarrow \{1110\}$: server $3$ goes into service when a type $1a$ or $2a'$ call arrives at sub-atom $1a$ or $2a$, or a type $2b$ or $3b$ call arrives at sub-atom $1b$ or $2b$.
- $\{0110\} \rightarrow \{1110\}$: server $1$ goes into service when a type $1b$ call arrives at sub-atom $1b$, or a type $2b$ or $3b$ call arrives at sub-atoms $1b$ or $2b$, and find only one server from their dispatching list.
- $\{1111\} \rightarrow \{1111\}$: server $4$ finishes service.

3.2. Model 2 – multiple dispatch and partial backup hypercub model considering the third status for each server

Model 2 extends model 1, taking into account not only events involving servers busy servicing emergency calls outside their bases, but also the eventual servers assistance to users requesting service at their bases (i.e., walk-in callers). Therefore, model 2 uses a third status for each server in order to represent separately its status of servicing a call at its base. A similar extension to the single dispatch hypercub model was studied by Larson and Mcknew (1982) to analyze a police deployment system. This study used a third status for a police patrol unit in order to represent the busy status on patrol initiated activities (called PIAs), which correspond to calls undertaken by the patrol officer as a result of something illegal that he or she sees from the patrol car (note that this is not a service assigned by the operations center).

In the present paper, we classify the calls serviced at the server’s base as type 0 calls (meaning that there is no server dispatch), with arrival rate $\lambda_j^{[0]}$ at atom $j$. The total arrival rate at atom $j$ then becomes: $\lambda_j = \lambda_j^{[0]} + \lambda_j^{[I]} + \lambda_j^{[II]}$. At a given time instant, the status of each server can be (0) idle, (1) busy responding to type $1$, $2$ or $3$ calls along the highway, and (2) busy responding to type 0 calls at its base. Consequently, there are $3^3$ possible states of the system. As aforementioned, type 0 calls have null travel time and no backup server. Model 2 assumes different mean service time to type 0 calls, therefore, each server $i$ has the mean service rate $\mu_i^{[I]}$ to type 1–3 calls and $\mu_i^{[II]}$ to type 0 calls. The superscripts [I] and [II] refers to the server status 1 and 2, respectively.

3.2.1. Equilibrium probability equations of model 2

We use the same example above with $N_j = 3$ atoms and $N = 4$ servers to illustrate how the equilibrium probability equations are formulated in model 2. For this four-server system, we have $3^4 = 81$ possible states, since each server has three statuses. Utilizing the dispatching preference list of Table 1 and considering that type 0 calls are only serviced by the base’s ambulance, the equilibrium equation to state $(1210)$, for instance, is defined as

$$
(\lambda_{1a}^{[1]} + \lambda_{2a}^{[2a']} + \lambda_{3a}^{[3b]} + \lambda_{1a}^{[3a]} + \lambda_{2a}^{[4]} + \lambda_{3a}^{[5]} + \mu_1^{[I]} + \mu_2^{[II]} + \mu_3^{[II]} \cdot P_{(1210)}
= (\lambda_{1b}^{[2b]} + \lambda_{1b}^{[2b]} + \lambda_{1b}^{[3b]} + \lambda_{2b}^{[2]} \cdot P_{(0220)} + \lambda_{4a}^{[4]} \cdot P_{(0101)} + (\lambda_{1b}^{[1]} + \lambda_{2a}^{[2a']} + \lambda_{1b}^{[2b]} + \lambda_{1b}^{[3b]} + \lambda_{1b}^{[4]} + \lambda_{2a}^{[2a']}) \cdot P_{(0210)} + \mu_1^{[II]} \cdot P_{(1211)} + \mu_3^{[I]} \cdot P_{(1212)}. \tag{2}
$$
According to Table 1, the transitions out of state \{1210\} (left side of equation) can occur when type 0, 1a, 2a' and 3a calls arrive at sub-atom 3a, type 2b calls arrive at sub-atom 3b, and type 2a'' calls arrive at sub-atom 2a, since only server 4 is idle. Note that, in state \{1210\}, server 2 is busy servicing type 0 calls and it goes out of service with rate $\mu_2^{[II]}$ (transition \{1210\} $\rightarrow$ \{1010\}), whereas, servers 1 and 3 go out of service with rates $\mu_1^{[II]}$ and $\mu_3^{[II]}$ (transitions \{1210\} $\rightarrow$ \{0210\} and \{1210\} $\rightarrow$ \{1220\}, respectively). The transitions into state \{1210\} (right side of the equation) are:

- $\{0200\} \rightarrow \{1210\}$: servers 1 and 2 go into service when a type 2b or 3b call arrives at sub-atom 1b or 2b, and one of its two preferential servers is busy (then, the third nearest server can also be dispatched).
- $\{1010\} \rightarrow \{1210\}$: server 2 goes into service (status 2) when a type 0 call arrives at sub-atom 1a (as mentioned, only its preferential server (2) can respond to this call).
- $\{1200\} \rightarrow \{1210\}$: server 3 goes into service (status 1) when a type 1a or 2a' call arrives at sub-atoms 1a or 2a, or a type 2b or 3b call arrives at sub-atom 1b or 2b and finds only one server from their dispatching list.
- $\{0210\} \rightarrow \{1210\}$: server 1 goes into service (status 1) when a type 1b call arrives at sub-atom 1b, or a type 2b or 3b call arrives at sub-atom 1b or 2b and finds only one server idle from their dispatching list (single dispatch).
- $\{1211\} \rightarrow \{1210\}$: server 4 finishes service (with rate $\mu_4^{[II]}$).
- $\{1212\} \rightarrow \{1210\}$: server 4 finishes service (with rate $\mu_4^{[II]}$).

3.3. Performance measures calculated by the models

A number of interesting performance measures can be evaluated by models 1 and 2 based on the equilibrium probabilities, such as, among others: loss probability of each type of call; workload of each server, distinguishing the emergency responses (types 1a, 2a', 2a'', 2b, 3a and 3b calls) from the responses at the base (type 0 calls); mean fraction of dispatches from each server’s base to each region; mean fraction of dispatches according to the type of call; mean region-wide travel time (considering all types of calls); mean travel time to each type of call; mean travel time of the two servers to type 2 calls; mean travel time to type 3 calls serviced by a double dispatch (when only two servers of the dispatching list are available); mean travel time of the first, second and third arriving server (in a triple dispatch) at the call location. For the sake of illustration, we present the evaluation of some of these performance measures below. More details of the definition and computation of these measures can be found in Iannoni (2005).

3.3.1. Loss probability

This is an important performance measure of EMS on highways, considering that a call can be lost to the system, even if there are servers available (because of partial backup). As a result of the multiple dispatching of servers, we may have: loss probability of any call ($P_p$) and loss probability of each type of call (e.g., the loss probability of type $m$ call ($P_p^{[m]}$), $m = 0, 1$ (1a and 1b), 2 (2a', 2a'', 2b), 3 (3a and 3b)). For instance, $P_p$ and $P_p^{[1]}$ are evaluated by

$$P_p = \sum_B \left( \frac{\sum_{j \in N^B} \bar{\lambda}_j}{\bar{\lambda}} \right) P_B$$

and

$$P_p^{[1]} = \sum_B \left( \frac{\sum_{j \in N^B} \bar{\lambda}_j^{[1]}}{\bar{\lambda}^{[1]}} \right) P_B,$$

where $B$ represents each state of the system (e.g., $B = \{1110\}$), $N^B$ is the set of atoms with all servers of its dispatching preference list busy at state $B$, $P_B$ is the equilibrium probability of state $B$ (obtained solving the equilibrium equations described above). Note that $\bar{\lambda}_j$ is the total arrival rate at atom $j$, $\bar{\lambda} = \sum_{j=1}^{N_A} \bar{\lambda}_j$ is the total arrival rate at the system, $\bar{\lambda}_j^{[1]}$ is the arrival rate of type 1 calls at atom $j$, and $\bar{\lambda}^{[1]} = \sum_{j=1}^{N_A} \bar{\lambda}_j^{[1]}$ is the total arrival rate of type 1 calls at the system.

3.3.2. Workloads of servers

The workloads $\rho_i^{[B]}$ (fraction of time that server $i$ is busy at status (1)) and $\rho_i^{[III]}$ (fraction of time the server $i$ is busy at status (2)) are calculated by taking into account all states that server $i$ is busy at status (1) and (2), respectively, that is
\[ \rho_i^{[I]} = \sum_{b \in D_i^{[I]}} P_B \quad \text{and} \quad \rho_i^{[II]} = \sum_{b \in D_i^{[II]}} P_B, \]  
\[ \text{where } D_i^{[I]} \text{ and } D_i^{[II]} \text{ are the set of states where server } i \text{ is busy at status (1) and (2), respectively.} \]

### 3.3.3. Fraction of dispatches

Models 1 and 2 can produce different fractions of dispatch statistics, such as

- **Fraction of all type 1 dispatches** (type 1a and 1b calls) that server \( i \) is sent to atom \( j \):
  \[ f_{ij}^{[1]} = \frac{(\lambda_j^{[1]}/A_j^{[1]})\sum_{b \in E_{ij}} P_B}{(1 - P_p^{[1]}),} \]
  \[ \text{where } E_{ij} \text{ corresponds to the set of states in which server } i \text{ is the first available server in the atom } j \text{ preferential list.} \]

- **Fraction of all type 2 dispatches** (type 2a, 2a', 2a'', and 2b calls) in which servers \( i \) and \( k \) are sent to atom \( j \):
  \[ f_{(i,k)j}^{[2]} = \frac{(\lambda_i^{[2]}/A_i^{[2]})\sum_{b \in E_{(i,k)j}} P_B}{(1 - P_p^{[2]}),} \]
  \[ \text{where } E_{(i,k)j} \text{ corresponds to the set of states in which servers } i \text{ and } k \text{ are two available servers in the atom } j \text{ preferential list, } A_i^{[2]} \text{ is the arrival rate of type 2 calls at atom } j, \text{ and } \lambda_i^{[2]} = \sum_{i,j} A_i^{[2][j]} \text{. For example, for the system example of Fig. 1, } f_{(2,3)j}^{[2]} \text{ is calculated by taking into account all type 2 dispatches in which servers 2 and 3 are sent to atom 1 (e.g., type 2a' calls at sub-atom 1a and type 2b calls at sub-atom 1b).} \]

- **Fraction of all type 2 dispatches** (type 2a', 2a'', 2b and 2b calls) in which server \( i \) is the single server sent to atom \( j \), due to either server \( k \) (for type 2a' calls) or servers \( k \) and \( l \) (for type 2a'' and 2b calls) being busy:
  \[ f_{ij}^{[2]} = \frac{(\lambda_i^{[2]}/A_i^{[2]})\sum_{b \in F_{ij}} P_B}{(1 - P_p^{[2]}),} \]
  \[ \text{where } F_{ij} \text{ corresponds to the set of states in which only server } i \text{ can service calls from atom } j \text{. For instance, for the system example of Fig. 1, } f_{ij}^{[2]} \text{ is calculated by taking into account all dispatches to type 2 calls, in which service server 2 is the single server sent to atom 1 (e.g., type 2a' calls at sub-atom 1a, when server 3 is busy, and type 2b calls at sub-atom 1b, when servers 1 and 3 are busy).} \]

Note that \( \sum_{j=1}^{N_A} \sum_{i=1}^{N_A} f_{ij}^{[1]} = 1 \) and \( \sum_{j=1}^{N_A} \sum_{i=1}^{N_A} \left[ \sum_{j=1}^{N_A} f_{ij}^{[2]} + \sum_{i=1}^{N_A} \sum_{j=1}^{N_A} f_{(i,k)j}^{[2]} \right] = 1 \). Other fractions of dispatch statistics can be calculated, such as the fraction of all dispatches that send server \( i \) to service any atom \( j \) call \( (f_{ij}) \); fraction of all dispatches that send server \( i \) to service atom \( j \) calls of type 1 (1a or 1b), 2 (2a', 2a'' and 2b) and 3 (3a and 3b) = e.g., \( f_{ij}^{[1]} \), \( f_{ij}^{[2]} \) and \( f_{ij}^{[3]} \) for single dispatch to type 1–3 calls, respectively; fraction of all dispatches that send server \( i \) and \( k \) to service atom \( j \) calls of types 2 and 3 (\( f_{(i,k)j}^{[2]} \) and \( f_{(i,k)j}^{[3]} \) for double dispatch to type 2 and 3 calls, respectively); and the fraction of all dispatches that sends server \( i \), \( k \) and \( l \) to atom \( j \) to service type 3 calls (\( f_{(i,k,l)j}^{[3]} \)), among others. For example:

\[ f_{(i,k)j}^{[2]} = \frac{(\lambda_j^{[2]}/A_j^{[2]})\sum_{b \in F_{(i,k)j}} P_B}{(1 - P_p^{[2]}),} \]

where \( \lambda \) is the total arrival rate at the system.

### 3.3.4. Aggregated measures of travel times

Using the fractions of dispatches, we can obtain some interesting mean travel time measures. It is worth mentioning that the travel time equations for model 1 are also valid for model 2, since for type 0 calls the travel time is null. For example:
• Mean travel time to type 1 calls:
\[
T^{[1]} = \sum_{j=1}^{N_d} \sum_{i=1}^{N} f^{[1]}_{ij} t_{ij},
\]  
(9)

where \( t_{ij} \) corresponds to the mean travel time from server \( i \) base to atom \( j \).

As in the study of Chelst and Barlach (1981), in the following expressions we used \( \min(t_{ij}, \bar{t}_{ij}) \) and \( \max(t_{ij}, \bar{t}_{ij}) \), instead of \( E(\min(t_{ij}, t_{kj})) \) and \( E(\max(t_{ij}, t_{kj})) \), respectively, since these approximations simplify the computation of the measures. Otherwise we should derive the probability distributions of \( t_{ij} \) to compute the expressions using \( E(\min(t_{ij}, t_{kj})) \) or \( E(\max(t_{ij}, t_{kj})) \), and, in the present study, this derivation would not be reasonable given that we have only a few observations of \( t_{ij} \) in the sample data. It is worth mentioning that, when the variance of \( t_{ij} \) is relatively small, which it is the case of the observations from our sample data, the approximations above would be likely more accurate.

• Mean travel time to type 2 calls (first arriving server for a type 2 dispatch):
\[
T^{[2]} = \sum_{j=1}^{N_d} \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} f^{[2]}_{(i,k)j} \min(t_{ij}, \bar{t}_{kj}) + \sum_{i=1}^{N} f^{[2]}_{ij} t_{ij}
\],
(10)

where \( f^{[2]}_{(i,k)j} \) and \( f^{[2]}_{ij} \) are the fractions of dispatches to type 2 calls, as defined before.

• Mean travel time for the two servers for a type 2 dispatch:
\[
T^{[2]}_{i} = \sum_{j=1}^{N_d} \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} f^{[2]}_{(i,k)j} (t_{ij} + \bar{t}_{kj}) + \sum_{i=1}^{N} f^{[2]}_{ij} t_{ij}
\].
(11)

• Mean travel time for the first arriving server and mean travel time for the second arriving server for a type 2 dispatch, when servers \( i \) and \( k \) are sent:
\[
T_{F} = \sum_{j=1}^{N_d} \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} f^{[2]}_{(i,k)j} \min(t_{ij}, \bar{t}_{kj}) \frac{1}{\sum_{j=1}^{N_d} \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} f^{[2]}_{(i,k)j}},
\]

\[
T_{S} = \sum_{j=1}^{N_d} \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} f^{[2]}_{(i,k)j} \max(t_{ij}, \bar{t}_{kj}) \frac{1}{\sum_{j=1}^{N_d} \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} f^{[2]}_{(i,k)j}}.
\]

• Mean travel time for server \( i \) to type 1 calls and type 2 calls:
\[
\bar{T}_{U_{i}^{[1]}} = \sum_{j=1}^{N_d} f^{[1]}_{ij} t_{ij} \frac{1}{\sum_{j=1}^{N_d} f^{[1]}_{ij}},
\]
(14)

\[
\bar{T}_{U_{i}^{[2]}} = \sum_{j=1}^{N_d} f^{[2]}_{(i,k)j} t_{ij} + f^{[2]}_{ij} t_{ij} \frac{1}{\sum_{j=1}^{N_d} f^{[2]}_{(i,k)j} + f^{[2]}_{ij}}.
\]
(15)

• Mean region-wide travel time (considering type 1–3 calls):
\[
\bar{T} = \left( \sum_{j=1}^{N_d} \sum_{i=1}^{N} f^{[1]}_{ij} + f^{[2]}_{ij} + f^{[3]}_{ij} \right) \cdot \bar{t}_{ij} + \left( \sum_{j=1}^{N_d} \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} f^{[2]}_{(i,k)j} + f^{[3]}_{(i,k)j} \right) \cdot \min(t_{ij}, \bar{t}_{kj})
\]
\[
+ \left( \sum_{j=1}^{N_d} \sum_{i=1}^{N-2} \sum_{k=i+1}^{N-1} \sum_{l=k+1}^{N} f^{[3]}_{(i,k,l)j} \right) \cdot \min(t_{ij}, \bar{t}_{kj}, \bar{t}_{lj}),
\]
(16)

where \( f^{[1]}_{ij}, f^{[2]}_{ij}, f^{[3]}_{ij}, f^{[2]}_{(i,k)j}, f^{[3]}_{(i,k)j} \) and \( f^{[3]}_{(i,k,l)j} \) are the fractions of dispatch to type 1–3 calls considering all dispatches in the system (as mentioned before).

We can use the models to obtain further performance measures, such as mean travel time for single dispatches in response to type 3 calls (in case only one server in the preference list is available); the mean travel
time for double dispatches in response to type 3 calls (in case only two servers in the preference list are available); the mean travel time for the first, second, and third servers, arriving at a type 3 call location; the mean travel time of all servers sent to type 3 location, among others.

4. Computational results

For the application of the hypercube models 1 and 2 presented in Section 3, we divided the highways into \( N_A = 8 \) atoms (segments), according to the primary area of each base established by the managers and operators of this system (see Fig. 1). The data were collected at the operations center within the period of January–September 2004, from the 1498 events recorded. We subdivided each atom of the system into layers \( a \) and \( b \), in order to consider the different dispatching preference lists according to the type of calls.

4.1. Results of model 1

Table 2 shows the 8 atoms subdivided into 16 sub-atoms (note that certain call types do not arise in certain sub-atoms, for example, there are no type 1b, 2b and 3b calls in sub-atom 8b because of its distance from the medical vehicle located in atom 1), and the dispatching preference list to each sub-atom. The data were collected at the operations center within the period of January–September 2004, from the 1498 events recorded. We subdivided each atom of the system into layers \( a \) and \( b \), in order to consider the different dispatching preference lists according to the type of calls.

We applied goodness-of-fit tests to verify the assumption of Poisson arrival process. For all atoms, the tests were unable to reject this assumption with 5% of significance. On the other hand, the goodness-of-fit tests rejected the assumption that the service processes are exponentially distributed. However, as discussed in Section 3, since the system does not allow for queuing, reasonable deviations from this assumption do not alter the accuracy of the model. We also applied analysis of variance to verify the difference among the servers’ mean service rates. This analysis indicated that, with 5% of significance, these means differ and the servers should be considered non-homogeneous for the model application (e.g., the mean service rates \( \mu_i \) are distinct).

The data related to the travel times of each server to each atom was also obtained from the sample and used to determine the mean travel time within the system.

The equilibrium probability equation for each state is formulated as described in Section 3. Given that the EMS analyzed has \( N = 6 \) servers (one medical vehicle and five rescue ambulances), there are \( 2^6 = 64 \) possible states. Using the preference list of Table 2, the equilibrium equations can be formulated as the equation below, which illustrates for state \( B = \{111000\} \) (with ambulances 1–3 busy and ambulances 4–6 idle):

<table>
<thead>
<tr>
<th>Atoms</th>
<th>Sub-atoms</th>
<th>Type of call</th>
<th>First server</th>
<th>Second server</th>
<th>Third server</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1a</td>
<td>la, 2a'</td>
<td>2</td>
<td>3</td>
<td>–</td>
</tr>
<tr>
<td>2a</td>
<td>1a, 2a'</td>
<td>2</td>
<td>3</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>2b</td>
<td>1b, 2b, 3b</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3a</td>
<td>1a, 2a'</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>–</td>
</tr>
<tr>
<td>3b</td>
<td>2b, 3b</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>–</td>
</tr>
<tr>
<td>4a</td>
<td>1a, 2a'</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>–</td>
</tr>
<tr>
<td>4b</td>
<td>2b</td>
<td>1</td>
<td>2</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>5a</td>
<td>1a, 2a'</td>
<td>4</td>
<td>5</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>5b</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>6a</td>
<td>1a, 2a'</td>
<td>5</td>
<td>4</td>
<td>5</td>
<td>–</td>
</tr>
<tr>
<td>6b</td>
<td>2b</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>–</td>
</tr>
<tr>
<td>7a</td>
<td>1a, 2a', 2a''</td>
<td>5</td>
<td>6</td>
<td>4</td>
<td>–</td>
</tr>
<tr>
<td>7b</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>8a</td>
<td>1a, 2a'</td>
<td>6</td>
<td>5</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>8b</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>
Substituting any of the 64 equations by the normalizing equation \( \sum_\theta P_\theta = 1 \) (the sum of all state probabilities must be equal to 1), we obtain a determined system with 64 linearly independent equations. The results of the state probabilities showed that the system stays with all servers available most of the time, given that \( P_{\{111111\}} \approx 0 \) and \( P_{\{000000\}} = 0.7964 \). Calculating the workload for each server \( i \), the results obtained by model 1 are \( \rho_1 = 0.0454 \), \( \rho_2 = 0.0621 \), \( \rho_3 = 0.0576 \), \( \rho_4 = 0.0226 \), \( \rho_5 = 0.0336 \) and \( \rho_6 = 0.0211 \). These results present relatively small deviations to the results from the sample analysis of Table 3 (see the last column). The system loss probability for all calls (\( P_p \)) is 0.57%; for type 1 calls (\( P_p^{[1]} \)) the loss probability is 0.63%; for type 2 calls (\( P_p^{[2]} \)), it is 0.23%; and for type 3 calls (\( P_p^{[3]} \)), it is 0.19%.

As described in Section 3, we can define several frequency statistics, according to the type of call in the system, such as single dispatch frequencies to type 1 (types 1a and 1b) calls; double dispatch frequencies to type 2 calls (types 2a, 2a' and 2b); double dispatch frequencies to type 3 calls (when one of the three candidate servers is busy); single dispatch frequencies to type 2a' calls (when one of the two candidate servers is busy) and to type 2a", 2b and 3 calls (when two of the three candidates servers are busy); and triple dispatch frequencies to type 3 calls. Calculating these statistics, we find that dispatches sending server 1 (medical vehicle) and 2 to atom 1 and server 3 to atom 2, present the highest type 1 dispatch frequencies (\( f_{j_1}^{[1]} \)): \( f_{11}^{[1]} = 15.43\% \), \( f_{21}^{[1]} = 16.66\% \) and \( f_{31}^{[1]} = 26.05\% \) (where \( \sum_{i=1}^{8} \sum_{j=6}^{8} f_{ij}^{[1]} = 1 \)). These dispatch frequency statistics obtained by the sample data analysis (using the relative error) were \( f_{11}^{[1]} = 14.61\% \) (i.e., a deviation of 5.6% from the value above), \( f_{21}^{[1]} = 17.26\% \) (3.5%) and \( f_{31}^{[1]} = 27.26\% \) (4.4%). We also compared other results for type 1 dispatch frequencies (\( f_{ij}^{[1]} \)) to the results from the sample data analysis and the mean deviation was relatively small.

Regarding type 2 dispatch frequencies (\( f_{ij}^{[2]} \), as defined in Section 3), for example, the fraction of dispatches sending servers 1 (medical vehicle), 2 and 3 to atom 1 are \( f_{11,12}^{[2]} = 56.67\% \), \( f_{11,13}^{[2]} = 2.73\% \), \( f_{11,2,3}^{[2]} = 4.42\% \),...
For the application of model 2, we analyze individually type 0 calls (walk-in callers), and type 1–3 calls (calls along the highway) as in Table 2. Table 6 presents the total arrival rates \( \dot{\lambda}_j \) (calls/h) of each atom \( j \) (which are the same as the ones of Table 3), the arrival rates of each sub-atom for call types 0, 1a, 1b, 2a′, 2a′′, 2b, 3a and 3b, and the distinct service rates: \( \mu_{i,[1]} \) (to type 1–3 calls) and \( \mu_{i,[0]} \) (to type 0 calls) for each server \( i \) of the system.

According to the discussion of Section 3.2, model 2 assumes \( 3^6 = 729 \) possible states of the system \((N = 6 \text{ servers})\). Calculating the state probabilities of model 2, we can obtain some additional performance measures by taking into account type 0 calls occurring at each atom. For example, we calculated the workloads for each server \( i \) in response to type 1–3 calls \( \rho_{i,[1]} \) and type 0 calls \( \rho_{i,[0]} \) separately. The results are shown in Table 7. As one would expect, the results for \( \rho_{i} \) obtained by model 1 are nearly the sum of the results for \( \rho_{i,[1]} \) and \( \rho_{i,[0]} \) obtained by model 2.

The travel time statistics calculated by model 2 to type 2 dispatches show relatively small deviations, compared to the results obtained by model 1 (the deviations related to the state probabilities are less than 0.05%). Next, we present some travel time measures related to the single dispatch of servers. For example, the mean travel time to locations of type 1 calls obtained by model 2 (as defined in Section 3.3) is \( \overline{T}_{1}^{[1]} \) = 8.746 min, and
the value of this measure obtained via sample analysis is $T_{\frac{1}{C_1}} = 8.762$ min (relative deviation of only $0.18\%$).

Recall from Table 4 that in model 1 we had $T_{\frac{1}{C_1}} = 7.033$, that is, $19.58\%$ less than the result obtained through model 2. Table 8 shows the travel time statistics for each server in response to type 1 calls ($TU_{\frac{1}{C_1}}$); such results were obtained in model 2, the sample data analysis and the relative deviation.

Observe in Table 8 that the mean travel time for each server of model 2 increases significantly if compared to the results calculated by model 1. It should be remembered that these time averages (model 2 and sample data) are computed without considering the type 0 calls (with null travel time). For example, for server 1 (medical vehicle), we found a deviation of more than $127\%$ from model 1 to model 2, since $57.8\%$ of the calls serviced by server 1 (medical vehicle) are of type 0 calls. As already pointed out, model 2 treated the call types 1–3 calls (with service rate $\mu_{\frac{1}{C_1}}^{(1)}$) and type 0 calls (with service rate $\mu_{\frac{1}{C_0}}^{(1)}$) individually. Therefore, although the current model is computationally more time-consuming than model 1, in the sense that the state space increases from $O(2^{N})$ to $O(3^{N})$, model 2 presents a much more accurate analysis of the real system.

Table 8
Travel time statistics for each server (in min) – single dispatch $TU_{\frac{1}{C_1}}$

<table>
<thead>
<tr>
<th>Server $i$</th>
<th>Model 2 $TU_{\frac{1}{C_1}}$</th>
<th>Sample $TU_{\frac{1}{C_1}}$</th>
<th>Deviation (%)</th>
<th>Model 1 $TU_{\frac{1}{C_1}}$</th>
<th>Deviation (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6.086</td>
<td>6.086</td>
<td>0.00</td>
<td>2.674</td>
<td>127.60</td>
</tr>
<tr>
<td>2</td>
<td>9.329</td>
<td>9.162</td>
<td>1.82</td>
<td>8.060</td>
<td>18.11</td>
</tr>
<tr>
<td>3</td>
<td>8.437</td>
<td>8.293</td>
<td>1.73</td>
<td>7.729</td>
<td>16.95</td>
</tr>
<tr>
<td>4</td>
<td>9.711</td>
<td>9.722</td>
<td>$-0.11$</td>
<td>7.771</td>
<td>21.85</td>
</tr>
<tr>
<td>5</td>
<td>10.155</td>
<td>10.324</td>
<td>$-2.02$</td>
<td>8.707</td>
<td>24.96</td>
</tr>
<tr>
<td>6</td>
<td>7.370</td>
<td>7.625</td>
<td>$-3.34$</td>
<td>6.356</td>
<td>15.95</td>
</tr>
</tbody>
</table>
5. Conclusions

In this study we show that the hypercube model can be adapted and applied to analyze EMS on highways with different types of calls (types 0–3) and servers (e.g., rescue ambulances and medical vehicle), considering particular dispatching policies such as zero-line capacity; partial backup; single and multiple (double and triple) dispatching of identical or distinct servers; servers responding to calls at their bases (e.g., users of the highways that stop at the server’s home station asking for help); among others. We present two model extensions (models 1 and 2), the difference between them being that the second considers a third status to each server, thus taking into account the busy time responding to calls at the server’s base; it also considers the different service times spent at such calls, as compared to the service times spent attending calls due to accidents along the highways.

These models were applied to a case study of an EMS operating on Brazilian highways, and the results were validated using analyses of the sample data obtained at the system’s operations center. These analyses showed that such methods are effective in evaluating important performance measures, such as the loss probabilities related to each different type of call, the frequency statistics for the calls and the aggregate travel time measures, distinguishing between the type and the number of servers dispatched to attend the call. Furthermore, we compared the results of the two models, showing that the model 2 analysis represents more accurately the functioning of the real system analyzed (by taking into account type 0 calls) – even though at the expense of an affordable additional computational effort (at least for systems of moderate size).

The extended hypercube models to analyze EMS on highways proposed in this study can be directly embedded into optimization procedures for ambulance deployment, for example, into a single node vertex substitution heuristic procedure, seeking to determine a set of ambulance locations which will maximize the expected coverage, as in Batta et al. (1989), Saydam and Aytug (2003), Chiyoshi et al. (2003) and Galvão et al. (2005), or into methods to determine optimal primary response areas for the ambulances’ location (districting problem). These models can also be applied to evaluate the trade-off among the conflicting performance measures of the system, such as mean region-wide travel time, servers’ unbalanced workloads, and the number of calls responded within a period of time. An interesting perspective for future research is to investigate these topics in actual situations.

Acknowledgements

The authors thank the two anonymous referees for their useful comments and suggestions. This research was partially supported by CNPq (Grants 140178/01-5, 522973/95-7) and Fapesp (Grant 05/53126-5).

References

Iannoni, A., 2005. Otimização da configuração e operação de sistemas médico emergencial em rodovias utilizando o modelo hipercubo. PhD Dissertation, Universidade Federal de São Carlos, Departamento de Engenharia de Produção, SãoCarlos, SP, Brazil.