Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds.

EMERGENCY MEDICAL SERVICE SYSTEM DESIGN EVALUATOR

Kyohong Shin
Inkyung Sung
Taesik Lee

Korea Advanced Institute of Science and Technology
291 Daehak-ro, Yuseong-gu
Daejeon, 305-701, REPUBLIC OF KOREA

ABSTRACT

Effectiveness of emergency medical services (EMS) depends on a wide range of decisions in its planning and operation phase such as ambulance locations and dispatching protocols. Much research has been conducted on EMS design and operational decision making in order to improve the quality of EMS systems. It is often the case that these research works focus on a decision problem on a specific aspect and tend to overlook possible interactions from other elements of an EMS system. This paper introduces a simulation model as a generic EMS system design evaluator, where a wide range of design and operational factors are comprehensively incorporated. Experiments using the developed model show that there exist interactions among many design and operational factors in an EMS system, which demonstrates the importance of considering all decisions when developing solutions for a specific decision problem in EMS design and operation.

1 INTRODUCTION

An emergency medical services (EMS) system is responsible for saving lives of emergency patients by providing first-aid on site and transporting a patient to a hospital for definitive care. Timely delivery of EMS is a critical factor for patient outcomes (Pons and Markovchick 2004). There is a large volume of research work that aims to improving the quality of EMS by minimizing its response time and service time (Su and Shih 2003, Peleg and Pliskin 2004). Response time is the time from the moment of arrival of a patient call until EMS providers arrive at a patient site. Service time is the time between a patient call arrival and the handover of the patient to a hospital (Andersson and Varbrand 2007).

A typical process of EMS operation is as follows: upon an arrival of a patient call, an ambulance is dispatched to respond to the call. A dispatched ambulance travels to arrive at the patient site and provides necessary first aid treatment on site. Then, it takes the patient to a destination hospital, usually the nearest emergency department, where the patient is admitted to receive appropriate medical care. Once the patient is safely admitted to an emergency department, the ambulance returns to its base station.

Several decisions are made in this EMS operation process. When a patient call arrives at an EMS call center, a dispatch decision is made to determine which of the available ambulances should be sent out to serve the call. After a dispatching decision is made, redeployment of the remaining ambulances is considered to obtain an optimal service coverage. In case of a mass casualty incident where many patients need EMS at the same time, priority for EMS provision needs to be determined. A destination hospital needs to be determined as well to ensure the patient receives necessary level of care in a timely fashion. Once an ambulance completes its assigned service task, a possible redeployment decision is considered again. Developing optimal solutions to these problems is an important task to improve the performance of an EMS system.

A common research method in EMS system design and operations involves simulation experiments. There are numerous EMS research articles that first develops alternative solutions to a decision problem and then subsequently tests the proposed solutions using a simulation model. Alternative solutions are typically developed by mathematical analyses or by intuitions and experiences of EMS experts. Simulation models are constructed to depict an EMS system, and the validity of proposed solutions are tested by simulation experiments.

In many cases, a simulation model used in such studies tends to focus on a specific phase or components within an EMS system that are directly relevant to the problem. In such simulation models, other elements in an EMS system are often treated by simplifying assumptions or are excluded from the model scope. For example, a simulation model for ambulance location problem may use a simple dispatching rule to send the nearest available ambulance to a patient location (Goldberg et al. 1990). It is also common to assume that upon completion of a patient transport, an ambulance always returns to its base station. Congestion at emergency departments may not be modeled, and a model always allows an ambulance to bring a patient to the nearest emergency department.

While such simulation models certainly simplify the model building and let analysts focus on the specific aspect of the EMS system, it may lead to some problems: it may fail to capture important interactions among EMS system elements (Sung and Lee 2012). These interactions can be significant enough to alter conclusions and recommendations from those simulation analyses. We extend the framework for an EMS system design evaluator proposed by Sung and Lee (2012) to build a simulation model that encompasses various components and decision factors in EMS system operations.

EMS system design evaluator presented in this paper is constructed based on the Activity Cycle Diagram (ACD) formalism. In the ACD formalism, a target system is modeled by interlinked activity cycles that entities and resources carry out. An individual activity cycle is constructed for each entity and resource, and these individual cycles are linked to represent the dynamics of an entire target system. Modeling based on the ACD formalism is convenient and intuitive especially when a system consists of clearly defined activity cycles by entities and resources. An EMS system is conveniently represented by a collection of activity cycles for patients and ambulances. In addition, decision factors in an EMS process correspond to a specific activity, thereby allowing to clearly define a decision module within a simulation model .

Using EMS system design evaluator, we conducted experiments to show potential interactions between the decisions made for different elements of an EMS system operation. Experimental results show that singling out a specific aspect of an EMS operation leads to a potentially misleading conclusion, which demonstrates the need to consider various elements and decision factors concurrently.

2 EMS SYSTEM DESIGN EVALUATOR

This section describes an overall architecture of EMS system design evaluator, and presents our ACD-based simulation model for an EMS system.

The goal of the proposed EMS system design evaluator is to allow EMS system analysts to consider various components and decision factors so that their solutions are assessed in the context of the entire system. Figure 1 shows a conceptual architecture of the EMS system design evaluator.

EMS system design evaluator consists of a main simulation model, data interface, and decision modules. A main simulation model is the ACD simulation model that simulates the operation of an EMS system. Data interface contains static and dynamic system data (i.e. system state variables). Static data includes initial configuration data and patient arrival data. They are specified by model users to configure the main simulation model. Dynamic data refers to a set of system state variables, that varies as a function of simulation clock. System state variables are updated either at the end of an activity execution or at a simulation update event. System data is accessed by decision modules as the main simulation model runs.

Data interface also contains operational decisions. Operational decision data is computed in response to a system state. Decision modules are decision logic functions external to the main simulation model, and they take necessary data from data interface to compute operational decisions. Decisions made from

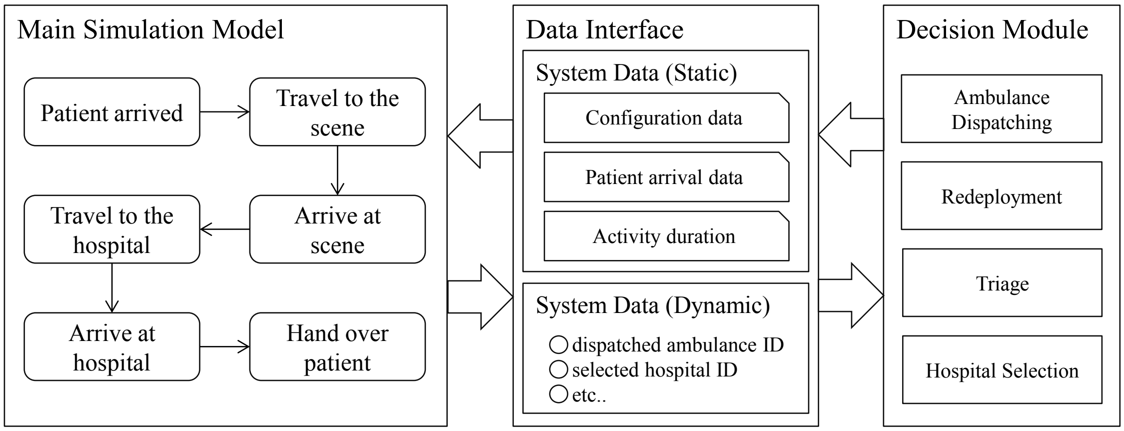


Figure 1: Conceptual architecture of EMS System Design Evaluator

decision modules are fed back to data interface, and the main simulation model refers to those decisions in subsequence execution.

Based on Sung and Lee (2012), four decision modules are included in the model: ambulance dispatching, redeployment, triage, and hospital selection. The main simulation model references operational decisions computed in the decision modules. For each decision module, only its outputs required from the main simulation model are defined, and EMS system analysts can build their own algorithm to develop a decision module. This architecture gives EMS system analysts a control over the specific decision/design problems they are interested in, while being able to use the rest of the EMS system simulation model including other decision components.

The rest of this section describes ACD modeling of an EMS system. We first briefly introduce the ACD formalism followed by a description of our reference EMS system. Then we present in detail the process of constructing the ACD-based EMS system model.

2.1 Activity Cycle Diagram

Activity Cycle Diagram (ACD) is a modeling formalism that represents a target system as a set of interacting activity cycles of entities and resources in the system (Paul 1993, Kang and Choi 2010). Tocher (1960) is credited for the development of ACD when he worked on a congestion problem of the steel plant. ACD is a convenient modeling approach especially when activities for entities and resources are clearly defined in a target system.

An activity cycle is formed by a sequence of alternating states, activity and queue, connected by an arc (Figure 2). An entity (or resource) proceeds along an activity cycle by moving from an activity state to a queue state, to the next activity state, etc. An activity state, denoted by a rectangle, literally refers to a state where an activity is carried out. A queue state, denoted by a circle, is a passive state that represents an entity or resource's waiting for a next activity. A basic condition for starting an activity is that all incoming queue states associated to the activity are non-empty. Each activity is associated with activity duration, which is the time it takes to complete an activity. Completion of an activity changes the system states, e.g., values of queue states, attributes of entity or resources, and dynamic variables. When a move from one state to the next requires specific conditions to be satisfied, an arc is annotated with a symbol $\sim$ to indicate the transition is conditional.

Each entity and resource in a target system has its activity cycle, which represents a temporal sequence of processes it goes through. An activity cycle is a closed cycle of activity and queue states in an alternating sequence. There always exists an activity that belongs to both an entity activity cycle and a resource activity cycle. These activities link entities and resources in a target system to make it a collection of interacting

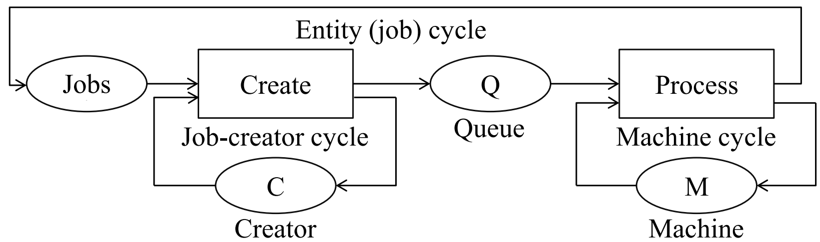


Figure 2: ACD model for a single queue system

cycles. Figure 2 shows an ACD model of a simple single queue system, where jobs (entity) arrive and receive a process from a machine (resource). For more details on ACD, refer to Kang and Choi (2010).

2.2 Reference Model: EMS System

A typical process of EMS system operation begins with an arrival of a patient call. When a patient call arrives at a call center, it searches for an available ambulance to dispatch to the scene. If all ambulances are busy, the patient is kept waiting. A dispatched ambulance travels to the patient site, locates the patient, provides necessary first aid, and boards the patient. It takes the patient to a destination hospital, and completes the patient handover process. Upon completing the patient handover, it returns to a base station. The ambulance may return to a base station different from where it has departed from. It may not return to a base at all, and directly head to another patient site if it is immediately tasked with a next service assignment.

In addition to the basic patient transport process described above, ambulance redeployment is another important operational feature in an EMS system. Ambulance redeployment refers to a practice of re-distributing available ambulances (i.e., ambulances on stand-by) from their current stations to different locations with a goal of maximizing the expected service coverage. It is triggered when the number of available ambulances either increases or decreases. For example, optimal locations of four ambulances are generally different from those when five ambulances are available. Thus, when one of the five ambulances is dispatched, it may be beneficial to redeploy the remaining four ambulances to achieve the optimal coverage by the four ambulances. Likewise, when the dispatched ambulance completes its task, increasing the total number of available ambulances back to five, the five ambulances may be redeployed to the optimal locations of a five-ambulance case.

2.3 ACD Model for EMS System

We model our target system - EMS system - based on the ACD formalism. In an EMS system, there is one entity type, which is `patient`. For resources, we have `ambulance` and `hospital` (Emergency Department). Two additional resource types - `call generator` and `mass casualty generator` - are defined to model entity generation in the ACD modeling framework. An activity cycle for each of these five elements is first constructed, and then integrated to model the entire EMS system.

An activity cycle for `call generator`, `mass casualty generator`, and `hospital` simply consists of one activity and one queue as shown in Figure 3. Figure 3(a) shows a `call generator` cycle where a patient arrives with an interarrival time of τ_{arrive}. `mass casualty generator` cycle in Figure 3 is the same as `call generator` cycle except that multiple patients are generated in one arrival. `hospital` cycle shows that a hospital with multiple beds (indicated by dots in the queue) treats a patient for the duration of $\tau_{treatment}$. Argument m indicates that there are m hospitals in the model.

`patient` entity has the following activity cycle shown in Figure 4. A patient arrives with an inter-arrival time of τ_{arrive} (`Arrive` activity). Each patient is assigned its attribute: time of call, severity, location, and expected length of stay (LOS). A patient waits for an available ambulance in queue `B1(k)`. Argument `k`

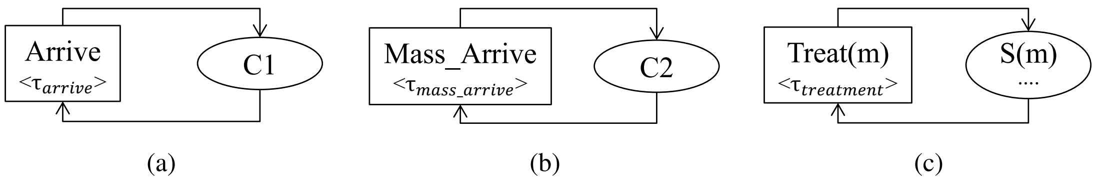


Figure 3: Activity cycle for simple resources: (a) `call generator` (b) `mass casualty generator` (c) `hospital`

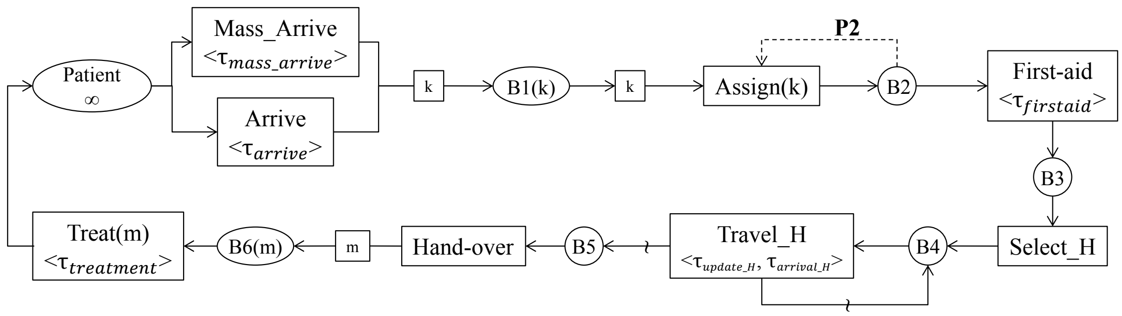


Figure 4: Activity cycle for `patient` entity

indicates the degree of patient's severity (1=low-severity, 2=high-severity). Then, an ambulance is assigned to the patient by `Assign(k)` activity. The patient waits for the ambulance to arrive in `B2` queue. If this patient is a low-severity patient, the originally assigned ambulance may cancel the assignment and be redirected to a newly arriving high-severity patient. In such a case, a new ambulance gets assigned to the patient. This is indicated by a dotted arc `P2`. When an ambulance arrives at the scene, the patient receives first-aid treatment for the expected duration of $\tau_{firstaid}$. A destination hospital is determined for the patient (`Select_H`), after which the patient is transported to a destination hospital by `Travel_H` activity. At a hospital, the patient is handed over to hospital staff (`Hand-over` activity), and when completed, the patient joins `B6` queue and the ambulance is released. Finally, the patient receives necessary treatment for the duration of $\tau_{treatment}$ (`Treat` activity). `m` in a square box is a parameter that defines a hospital index to avoid duplicated presentation of `Treat` activities for *m* units of hospitals. Patients leaving a hospital are discarded (i.e., leaves the EMS system), which is represented by moving to an entity pool queue, `Patient` ∞ queue.

`Travel_H` activity has two durations, τ_{update_H} and $\tau_{arrival\_H}$. This is necessary to track the current location of a patient (and the ambulance carrying the patient). τ_{update_H} is the time difference between the current simulation time and the next simulation update time. $\tau_{arrival\_H}$ is the time difference between the current simulation time and the expected arrival time at a destination hospital. If $\tau_{update\_H} < \tau_{arrival\_H}$, $\tau_{arrival\_H}$ is decreased by τ_{update_H}; patient location is updated after τ_{update_H} by using an average road travel speed at the moment; and the patient moves to `B4` queue and back to `Travel_H` activity. If $\tau_{update\_H} \geq \tau_{arrival\_H}$, a patient leaves `Travel_H` activity after $\tau_{arrival\_H}$ to move to `B5` queue.

Activity cycle for `ambulance` is slightly more complicated by two factors. First, a possible ambulance redeployment is included. Representing redeployment implies that dispatching one ambulance possibly triggers activities of other ambulances. Second, we assume that a busy ambulance is made available as soon as it hands over its patient to a hospital, as opposed to when it has returned to its base station. We also assume that an ambulance on its route to a redeployment destination is available to take on an incoming patient call. This requires to represent an interrupt type of mechanism in an activity cycle.

Figure 5 shows an activity cycle of `ambulance` resource. An available ambulance in `Amb` queue is assigned to a patient by `Assign(k)` activity. This ambulance travels to the patient site (`Travel_P`

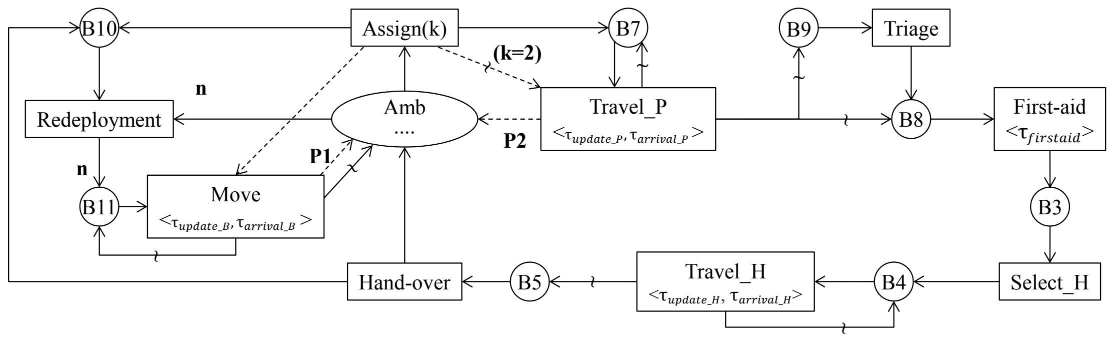


Figure 5: Activity cycle for `ambulance` resource

activity). `Travel_P` activity is similar to `Travel_H` activity in the patient activity cycle. It has two time durations, τ_{update_P} and $\tau_{arrival\_P}$, which are defined and used the same way as τ_{update_H} and $\tau_{arrival\_H}$. Location of the ambulance continuously updated by `Travel_P` activity. If this ambulance on its way to the originally assigned patient happens to be the nearest ambulance for a newly arriving high-severity patient, then `Assign(2)` activity sends a signal to `Travel_P` activity (dashed arc) and it is reassigned to the high-severity patient (arc `P2`). Once `Travel_P` activity is completed, the ambulance provides necessary first-aid treatment on scene, which is `First-aid` activity. In case of a mass casualty incident, where a large number of patients need EMS simultaneously, `Triage` activity is carried out to determine a highest priority patient to provide care to. After provision of a first-aid treatment, a destination hospital is determined (`Select_H`, and the ambulance brings the patient to a destination hospital (`Travel_H`). Upon arriving at a hospital, it hands the patient over to hospital staff (`Hand-over`).

When one ambulance is assigned (i.e., dispatched) to a service task by `Assign(k)` activity, the number of available ambulances in `Amb` queue is decreased by one, and `B10` queue is increased by one. Non-empty `B10` and `Amb` queue satisfies a triggering condition for `Redeployment` activity.

`Redeployment` activity can take place when the number of available ambulances changes, and there are two such cases: 1) an ambulance is dispatched to serve an incoming call (`Assign(k)`), and 2) an ambulance is made available by handing over its patient to a hospital (`Hand-over`). This is indicated by an incoming arc from `B10` queue, which has two incoming arcs from `Assign(k)` and `Hand-over`. Also, redeployment is feasible only when at least one ambulance is available. This is enforced by an incoming arc from `Amb` queue. Once a redeployment decision is made to move *n* ambulances from their current locations to new sites, they travel to the new sites (`Move`). `Move` activity works similarly to `Travel_H` to track ambulance's current location. Note that how many and which ambulances to move to which locations is a decision made by an execution of `Redeployment` activity. Also note that `Redeployment` activity may return a null-decision, i.e. stay at current location or return to its home base station.

Recall that our model allows an ambulance to be available as long as it does not currently carry a patient. To reflect this feature, an arc is drawn from `Hand-over` activity to `Amb` queue. Also, to make an ambulance available while moving, a dashed arc from `Assign(k)` to `Move` is used. When the ambulance is called for a service during its move, `Assign(k)` activity sends a signal to `Move` activity along the dashed arc and the ambulance is assigned to the patient (arc `P1`).

The five cycles presented in Figure 3, 4, 5 are now linked along their common activities to form an entire cycle of the EMS system. Figure 6 shows a complete ACD cycle.

3 EXPERIMENT

Experiments are conducted to investigate the effect of interactions between decisions for different parts of an EMS system operation. We model the real EMS system of the city of Bucheon in Korea, and use the EMS patient data of January 2010.

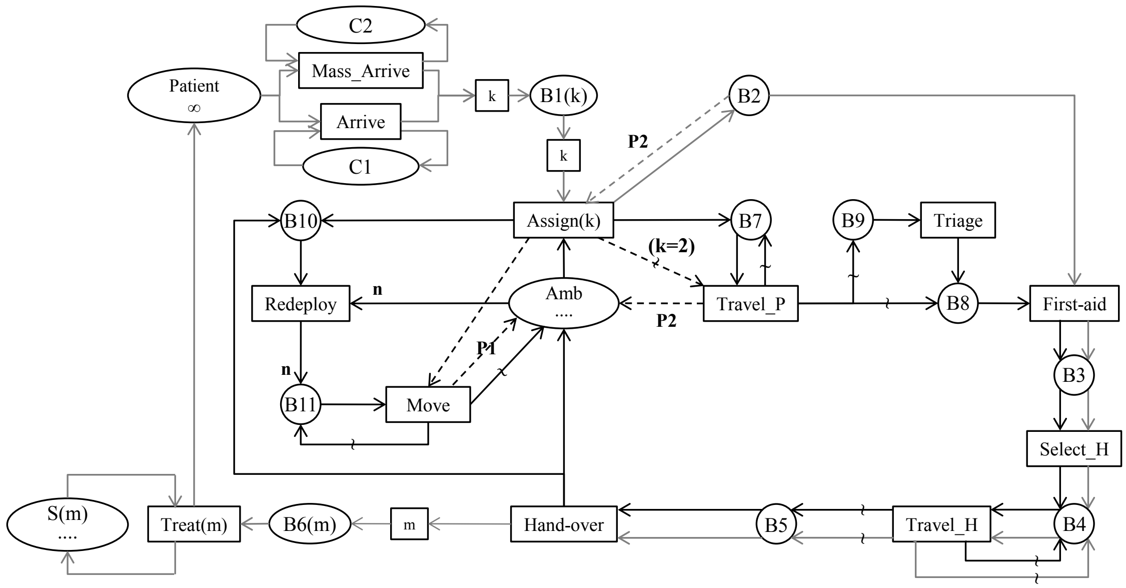


Figure 6: ACD model for an EMS system

3.1 Input Data for EMS System Model

This section describes the input data used to configure the model for the city of Bucheon in Korea. We collect actual road network data, locations of hospitals, locations of EMS base stations, and the number and default deployment distribution of ambulances to configure the simulation model. EMS log data for the month of January of 2010 is used to extract information on patient arrival and duration for each activity in the model.

Figure 7 shows the road network with EMS stations and hospitals indicated. A road network for the model is constructed with the city's road segments for which hourly traffic information is available. The road network is represented as a network of 46 nodes and 65 edges. Length of each edge is defined by the actual distance between the nodes, and average hourly travel velocity for each edge is available. There are 10 EMS base stations in the city, and each center hosts one ambulance. Thus, the default (i.e. as-is) deployment of the city's ambulances is one ambulance per station. Actual location of each EMS station is approximated to the nearest node of the model's road network. From the EMS log data, we chose 9 hospitals in the city that logged at least 10 patient visits during the month. These hospitals are classified into three levels by their size and the level of care they provide.

There are 12 activities in our EMS system model, and all but `Redeployment` and `Select_H` activities have non-zero activity duration: τ_{arrive}, τ_{mass_arrive}, τ_{assign}, τ_{arrive_P}, τ_{triage}, $\tau_{firstaid}$, τ_{arrive_H}, $\tau_{handover}$, τ_{treat}, and τ_{arrive_B}. Note that for the current version of our model, mass casualty incident and triage are not considered.

We use the EMS log data to extract activity duration information. EMS log data contains time information at six time stamp points: call arrival (t_{call}), ambulance departure at a station ($t_{depart\_station}$), patient contact ($t_{contact\_patient}$), departure from the patient site (t_{depart_scene}), arrival at a hospital ($t_{arrival\_ED}$), and return to the station (t_{return}). From the log's time stamp information, some of the activity durations are directly obtained as follows:

- τ_{arrive} = interarrival time of patient calls
- $\tau_{assign} = t_{depart\_station} - t_{call} \sim$ lognormal(2.19, 1.03)
- $\tau_{firstaid} = t_{depart\_scene} - t_{contact\_patient} \sim$ gamma(3.46, 2.57)

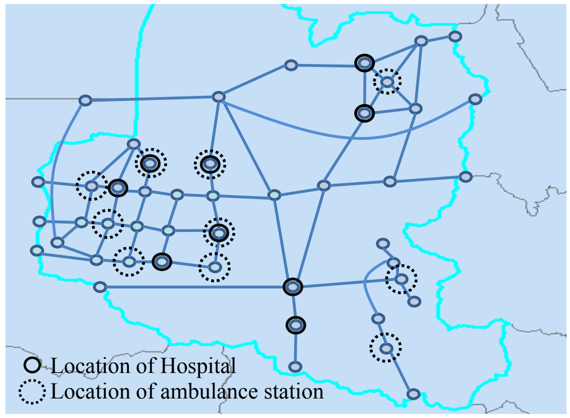


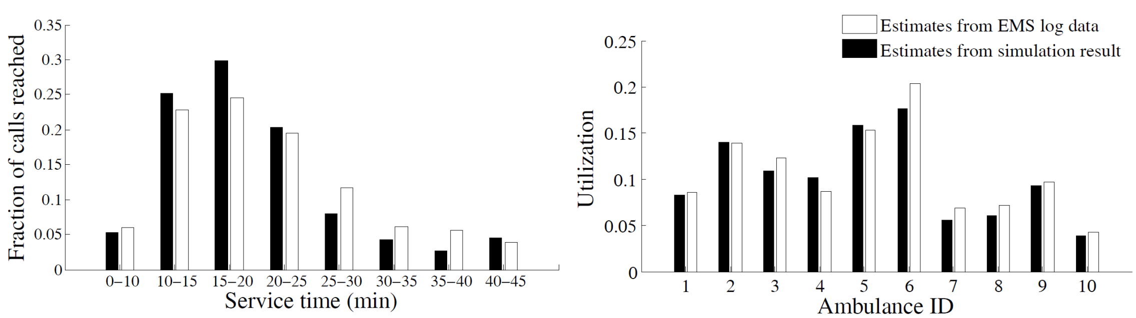
Figure 7: Road network used in the simulation model and the locations of EMS stations and hospitals

Since we use actual road network data with traffic information, if an origin and destination for a trip is given, we can estimate travel time component for τ_{travel_P}, τ_{travel_H} and τ_{move}. We first estimate τ_{travel_H} by the shortest distance divided by the average travel velocity, and it reasonably agree with $(t_{arrive\_ED} - t_{depart\_scene})$ from the EMS log data. Knowing that travel time estimation by distance/velocity gives a reasonable estimate for a travel component, we estimate $\tau_{handover}$ as

- $\tau_{handover} = (t_{return} - t_{arrive\_ED}) -$ (estimated travel time between the ED and return-station)

Lastly, we learned from interviews with EMS experts that $t_{contact\_patient} - t_{depart\_station}$ consists of sheer travel time and time to locate or reach a patient. Thus, we assume $\tau_{travel\_P} = \tau_{sheer\_travel\_P} + \tau_{locate\_patient} = t_{contact\_patient} - t_{depart\_station}$. With this assumption, we estimate $\tau_{locate\_patient}$. So, in the simulation model, τ_{travel_P} is obtained by adding the sheer travel time (computed by shortest distance divided by average travel velocity) and $\tau_{locate\_patient}$.

We validate our simulation model using the actual patient arrival data. Ambulances are dispatched to each patient according the nearest-available policy, and patients are taken to a hospital as specified in the EMS log data. Figure 8(a) compares service time from the simulation model and the EMS log, which shows a good agreement. We also measure utilization of 10 ambulances from the simulation model and from the EMS log, and it shows a good agreement as well. Thus, we conclude that our simulation model represents the target EMS system with acceptable fidelity.



(a) (b)

Figure 8: Comparison between simulated results and EMS log data: (a) service time (b) ambulance utilization

3.2 Experimental Setting

We test interactions between decision policies for three decision modules: ambulance dispatching, hospital selection, and redeployment. These decision modules are incorporated in the simulation model as `Assign`, `Select_H`, and `Redeployment` activity. We use two distinct decision policies for each decision module. Table 1 summarizes the experimental setting.

For ambulance dispatching, one policy is a nearest-available policy. Upon an arrival of a patient call, this policy searches for currently available ambulances and assigns the ambulance nearest to the patient. The other policy is a likelihood-based policy, which was proposed by Repede and Bernardo (1994). When a patient call arrives, it searches for the nearest available ambulance. Then it computes the expected time for the ambulance to arrive at the scene. If the ambulance is expected to arrive within a given time standard, this ambulance is sent to the patient. If not, then this policy dispatches an ambulance from a station that has the least likelihood of a new arrival of a nearby patient. For the time standard, we use 8-minute for high-severity patient and 14-minute for the rest. In both policies, an ambulance on route to a low-severity patient can be re-assigned and re-directed to a new arrival of high-severity patient if the ambulance on the move happens to be the nearest ambulance to the high-severity patient.

For hospital selection, we use a nearest-available policy and a preferential-selection policy. A nearest-available policy searches for hospitals currently under a capacity limit, and simply chooses the nearest hospital. Alternatively, we use a preferential-selection policy. A preferential policy reflects the fact, observed from the EMS log data, that not all patients are transported to the nearest hospital from the scene. From the EMS log, we obtain the number of EMS patients that each hospital in the city admitted, and use this fraction as a probability to choose a particular hospital.

Redeployment module uses two policies. One is a null-policy where redeployment is not practiced at all. Ambulances always return to their home station, and stay there. The other uses a policy proposed by Gendreau, Laporte, and Semet (2006). For a region with N ambulances, the policy uses a compliance table to shift locations of available ambulances. The compliance table shows a desired deployment configuration for k-ambulance case, where $k = 1,\ldots,N$. It is computed by solving a maximal expected coverage relocation problem (MECRP). Using Bucheon city data, we solve MECRP problem to construct a compliance table to use in the redeployment module.

Since the effectiveness of these policies may differ depending on the volume of EMS demand, we use five levels of demand volume for the experiment. Using the actual number of patients from the EMS log data (Jan. 2010) as a nominal case, patient arrival data of 50%, 80%, 100%, 120%, and 150% of the nominal case are generated. This is done by obtaining average patient arrival rate at each node from the EMS log data, and multiplying a scaling factor to the nominal rate. Patient arrival is assumed to follow Poisson process.

Table 1: Experimental Setting

Factor	Description	
Ambulance dispatching policy	(1) Nearest-available	(2) Likelihood-based
Hospital selection policy	(1) Nearest-available	(2) Preferential-selection
Redeployment policy	(1) Null	(2) MECRP
EMS demand volume	$\alpha\times$ nominal volume;	$\alpha = 0.5, 0.8, 1.0, 1.2, 1.5$

4 RESULT AND DISCUSSION

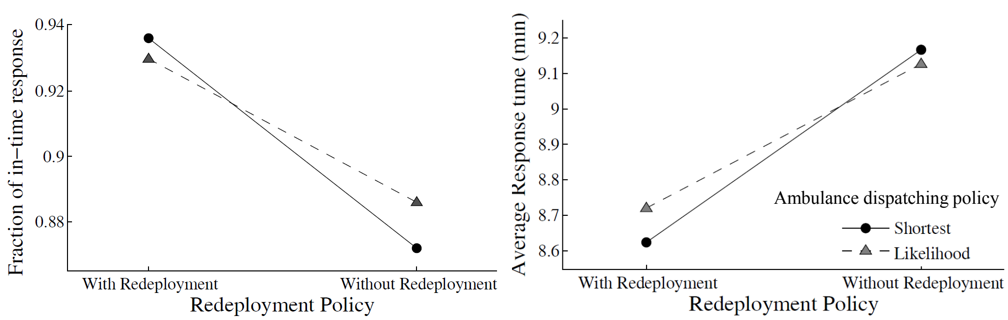
Total of 40 scenarios of different policy combinations and demand level are evaluated in the experiment. Simulation for each scenario is replicated 10 times. Length of a simulation run is 30 days, with a warm up period of one day. As key performance indicators, we examine two output variables: 1) a fraction of patient calls that an ambulance's response time is within the time standard, and 2) average response time for all patients calls. Response time is the time until an ambulance makes a contact with a patient, i.e. $t_{contact\_patient}$ - t_{call}. 4-way ANOVA test is conducted to examine possible interactions between ambulance assignment(X1), hospital selection(X2), redeployment policy(X3), and demand volume(X4).

Table 2 shows the results from a 4-way ANOVA test for the fraction of within-standard responses and average response time. In both cases, interaction effects are observed: ambulance dispatching policy and redeployment policy (X1*X3), hospital selection policy and redeployment policy (X2*X3), hospital selection policy and demand volume (X2*X4), and redeployment policy and demand volume (X3*X4).

Table 2: ANOVA test results. X1 = ambulance dispatching policy, X2 = hospital selection policy, X3 = redeployment policy, X4 = demand volume

	Fraction of within-standard responses					Average response time				
Source	Sum Sq.	d.f.	Mean Sq.	F	Prob>F	Sum Sq.	d.f.	Mean Sq.	F	Prob>F
X1	0.00003	1	0.00003	3.57	0.0597	0.666	1	0.6658	28.55	0
X2	0.03063	1	0.03063	366.44	0	11.962	1	11.9624	512.93	0
X3	0.25488	1	0.25488	3048.99	0	23.083	1	23.083	987.76	0
X4	0.4828	4	0.1207	1443.86	0	155.484	4	38.8709	1666.72	0
X1*X2	000003	1	0.00003	0.37	0.544	0.003	1	0.0035	0.15	0.6993
X1*X3	0.00571	1	0.00571	68.32	0	0.451	1	0.4509	19.34	0
X1*X4	0.00083	4	0.00021	2.49	0.0429	0.078	4	0.0194	0.83	0.5061
X2*X3	0.00216	1	0.00216	25.89	0	0.824	1	0.8237	35.32	0
X2*X4	0.01159	4	0.0029	34.67	0	3.643	4	0.9107	39.05	0
X3*X4	0.01237	4	0.00309	36.98	0	3.518	4	0.8795	37.71	0
Error	0.03152	377	0.00008			8.792	377	0.0233		
Total	0.83282	399				208.504	399			

Interactions between decision policies have implications on optimal choice of a policy for each decision module. Let us look at one of the interaction effects for more details. Figure 9 shows an interaction effect between ambulance dispatching policy and redeployment policy. In the graph, patient volume is nominal ($\alpha = 1.0$), and the nearest-available policy is used for hospital selection. Under MECRP redeployment policy, the nearest-available dispatching policy yields a better outcome than the likelihood-based policy in both the fraction of in-time response and average response time. On the other hand, under the null redeployment, we have opposite results. Thus, the choice of an optimal policy for ambulance dispatching is dependent on the type of redeployment policy used.



(a) (b)

Figure 9: Interaction effect between redeployment policy and ambulance dispatching policy: (a) fraction of in-time response (b) average response time

This interaction effect is more pronounced in the following example where we make an assumption regarding a redeployment procedure: an ambulance on the move to relocate to a new station is considered unavailable until it arrives at the new station. As shown in Figure 10, which of the two dispatching policies produces a better outcome depends on the type of redeployment policy. Furthermore, in this example, which deployment policy shows a better outcome also depends on the type of dispatching policy: under the nearest-available dispatching policy, a better choice is the MECRP policy. When the likelihood-based policy is used, it is the null redeployment policy.

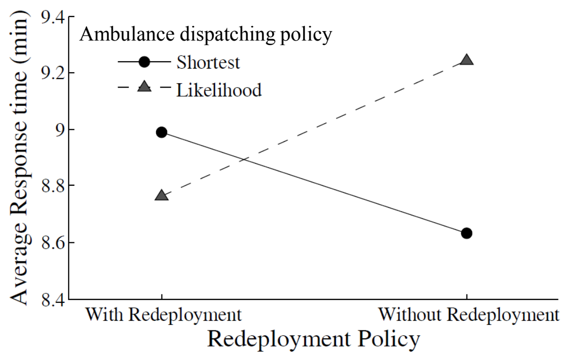


Figure 10: Interaction effect between redeployment policy and ambulance dispatching policy with additional, restrictive assumption

These examples demonstrate that the optimal choice of a policy for one decision component is affected by choices made for other decision components. Interaction effects can potentially be significant enough to lead to a completely opposite conclusion. Thus, evaluation of a specific component of EMS system design should explicitly consider the entire EMS system elements and various decisions associated with them.

5 CONCLUSION

Operating an EMS system involves many operational and planning decisions, and the quality of those decisions determine the level of service of an EMS system. When developing an optimal decision for each element of EMS system operation, it is important to take into account interaction effects among various decisions. Our experiments confirm that the existence of interaction effects among various decisions can be significant enough to affect the choice of an optimal decision for a system element.

This paper presents a prototype of EMS system design evaluator. A simulation model for a generic EMS operation is constructed based on the ACD formalism. Four decision modules - ambulance dispatching, hospital selection, triage, and redeployment - have been identified, and they are incorporated in the ACD model as individual activities. Decision modules are external to the simulation model, and model users can implement their own decision logic. A decision module accesses EMS system configuration data and system state variables, and returns its decisions to the simulation model through a set of output variables. This architecture gives EMS system analysts a control over the specific decision/design problem they are interested in, while being able to consider other decision elements. Our aim is to improve the simulation model presented in this paper and make it publicly available for use by EMS system researchers in near future.

ACKNOWLEDGMENTS

This research was supported by the Public welfare & Safety research program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0029883).

REFERENCES

Andersson, T., and P. Varbrand. 2007. "Decision Support Tools for Ambulance Dispatch and Relocation.". *Journal of the Operational Reseach Society* 58:195–201.

Gendreau, M., G. Laporte, and F. Semet. 2006. "The Maximal Expected Coverage Relocation Problem for Emergency Vehicles.". *Journal of the Operational Research Society* 57 (1): 22–28.

Goldberg, J., R. D. J. M. Chen, M. Mitwasi, T. Valenzuela, and E. Criss. 1990. "A Simulation Model for Evaluating a Set of Emergency Vehicle Base Locations: Development, Validation, and Usage.". *Socio-economic planning sciences* 24 (2): 125–141.

Kang, D., and B. K. Choi. 2010. "Visual Modeling and Simulation Toolkit for Activity Cycle Diagram.". In *Proceedings 24th European Conference on Modelling and Simulation*, edited by A. Bargiela, S. Crowley, and E. J. H. Krckhoffs.

Paul, R. J. 1993. "Activity Cycle Diagrams and the Three Phase Method.". In *Proceedings of the 1993 Winter Simulation Conference*, edited by G. W. Evans, M. Mollaghasemi, E. C. Russell, and W. Biles, 123–131. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Peleg, K., and J. S. Pliskin. 2004. "A Geographic Information System Simulation Model of EMS: Reducing Ambulance Response Time.". *American Journal of Emergency Medicine* 22:164–170.

Pons, P. T., and V. J. Markovchick. 2004. "Eight Minutes of Less: Does the Ambulance Response Time Guideline Impact Trauma Patient Outcome?". *Journal of Emergency Medicine* 22 (1): 164–170.

Repede, J. F., and J. J. Bernardo. 1994. "Case Study Developing and Validating a Decision Support System for Location Emergency Medical Vehicles in Louisville, Kentucky.". *European Journal of Operational Research* 75:567–581.

Su, S., and C.-L. Shih. 2003. "Modeling an Emergency Medical Services System Using Computer Simulation.". *International Journal of Medical Informatics* 72:57–72.

Sung, I., and T. Lee. 2012. "Modeling Requirements for An Emergency Medical Service System Design Evaluator.". In *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A. M. Uhrmacher. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Tocher, K. D. 1960. "An Integrated Project for the Design and Appraisal of Mechanized Decision-Making Control Systems.". *Journal of the Operational Research Society* 11 (1/2): 50–65.

AUTHOR BIOGRAPHIES

KYOHONG SHIN is a Master's student in the department of Industrial and Systems Engineering at KAIST, Republic of Korea. His current research interest focus on modeling and simulation of emergency medical service system. His email address is hong906@kaist.ac.kr.

INKYUNG SUNG is a Ph.D student in the department of Industrial and Systems Engineering at KAIST, Republic of Korea. His research interest is on modeling and simulation and its application to healthcare delivery system, EMS systems in particular. His email address is inkyung@kaist.ac.kr.

Dr. TAESIK LEE is an associate professor of Industrial & Systems Engineering Department at KAIST, Korea. His research interests are primarily in the area of system design, modeling and simulation, especially for large-scale, complex systems. For the past few years, he has been working on designing various aspects of healthcare delivery systems. Dr. Lee obtained his MS and PhD degree from MIT in the Dept. of Mechanical Engineering. His email address is taesik.lee@kaist.edu.