Near-Optimal Emergency Evacuation with Rescuer Allocation

Erol Gelenbe and Qing Han
Imperial College London
Department of Electrical and Electronic Engineering
Intelligent Systems and Networks Group
Email: {e.gelenbe,qing.han12}@imperial.ac.uk

***Abstract*—Emergency management systems are Cyber-Physical-Human Systems (CPHS) that use sensing, together with communications and control, to guide humans and physical systems such as vehicles, towards safe desirable outcomes in the shortest possible time. When human health and safety, and lives also, are at stake, it is important to take decisions in real-time with the best possible use of resources, including the critical resource of emergency personnel. Such distributed decision problems are so complex that the resulting optimisation and allocation problems can only be handled in a fast and timely manner using efficient heuristics. Thus in this paper we apply a recent resource allocation algorithm based on the Random Neural Network (RNN) to allocate rescuers to those evacuees whose health level has deteriorated beyond a certain level in the course of an evacuation. The approach is evaluated by simulating the evacuation of a three story building using the Distributed Building Evacuation Simulator (DBES) developed at Imperial College. The simulations show that the outcome of the evacuation can be significantly improved in this manner, in particular for larger numbers of evacuees.**

***Index Terms*—Cyber-physical systems, emergency management, random neural network, real-time decisions.**

I. Introduction

Cyber-Physical-Human Systems (CPHS) can use wireless technologies [1], sensing, and distributed decision-making and control [2], [3], [4] with ubiquitous computing to improve the outcomes for emergency management situations [5], [6] where situational assessment and location of victims are needed to guide the complex tasks of search and rescue [7], [8], [9]. Then, rescuers with appropriate assets can be dispatched in a manner which maximises the chances of saving the survivors, removing the injured, and evacuating everyone in a timely manner.

The core algorithmic problem in this area is the assignment of resources and rescuers to victims and emergency situations and areas, which is a very complex instance of the Assignment Problem [10], [11]. Similar problems arise in distributed computation [12] when a set of processors and other computational resources such as memory and input-output units must be assigned to a set of large programs. Both in emergency management and in distributed computation, the key information that is needed for an optimal assignment actually varies with time, and hence the optimum solution will also vary with time. As a consequence, the optimisation problem must be repeatedly solved, or handled as a dynamic control problem. This leads to major issues of computational complexity [13]. The routing of evacuees towards exits, and of rescuers towards evacuees and then towards exits also has many elements similar to network routing problems [14] which are often better addressed with "sensible" heuristics [15] rather than with optimal solutions. Capacity constrained routing approaches in this area, similar to the ones in vehicular traffic or telecommunications have also been considered [16], [17].

In an emergency, the behaviour of the evacuees cannot be known in advance because congestion and panic will also significantly shape human behaviour. Similarly, the emergency itself, for instance a fire, will move dynamically [18] and modify the conditions throughout time. In a distributed computation, many factors can only be known at run time so that (as in an emergency) the optimal allocation of resources must change over time.

This challenging and realistic problem is NP-hard when posed as a non-linear optimisation problem, and its computational difficulty is exacerbated by the fact that the allocation of resources and rescuers must be computed, or at least re-evaluated again, each time that conditions change, which is possible but impractical. Since resources which have been already committed would have to be reallocated, the algorithm may have to be run several times. Because of these challenges, many authors have considered the use of collaborative multi-agent systems as a way to construct dynamic heuristic solutions to this problem [19], [20], [21], [22].

In [23] a resource allocation method was developed assuming that once the situational assessment is done, the information is shared with all the assets or rescuers that are used to rescue the evacuees. This information is input into all of the copies of the identically trained RNN, which provides a unique set of instructions to all of the rescuers; these instructions can also cover other types of resources that rescuers may need. All of the rescuers then consult the *same "oracle"* based on a Random Neural Network (RNN) [24], [25], and follow its instructions. The RNN is previously trained using many real or simulated emergencies in the same physical context, so as to offer a near-optimal solution to a large number of emergencies which are similar to the many real situations that may be encountered. The resource allocations provided by the trained RNN are non-conflicting, so that an asset is never allocated to two different emergency sites. Thus *all* of rescuers will use their copy of the trained RNN to know which evacuees they should attend to, based on a situational assessment that is broadcast to all the assets [26], [27].

In this paper we will evaluate this approach for the evacuation of a specific building. The idea we pursue is to allocate rescuers to evacuees if the victims' health conditions deteriorate beyond a certain level. The approach is evaluated via simulations using the Distributed Building Evacuation Simulator (DBES) [28], which is a convenient agent based simulation tool which incorporate a building's lay-out and the initial locations of evacuees and rescuers, and which can also simulate the spread of hazards such as fires or noxious gases.

In the rest of this paper, Section II reviews related work, while Section III discusses the RNN based algorithm. The simulations are discussed in Section IV, and conclusions are drawn in Section V.

II. Related Work

Situational assessment [29] is the first stage of rescue, and requires smart sensing and decision making [9]. The rescuers may run into danger in hazardous sites, and may need to interact with each other. Such uncertainties can lead to inefficiency in resource allocations, while the overall urgency of such situations requires fast computational and human decisions. Research on emergency evacuation [30] must also consider that victims can be obstructed by hazards. Thus resources to be dispatched must provide services in a very efficient manner [31], [32], [33], [34].

There has been very substantial work on the use of wireless communications and sensor networks to alleaviate the hazards related to emergency evacuation [35], [36]. Related issues of navigation with the help of sensor networks [37], [38], [39] have also received substantial attention. Various studies have detailed how these techniques can be used during and emergency or fire [40], [41] and specific issues such as rendezvous planning with wireless sensor networks have also been considered [42].

Many studies have focused on the specific techniques that can be used during an emergency, rather than on the . of the techniques within a specific set of relevant emergencies. Thus tools such as the distributed building evacuation simulator [28] can reduce the simulation time required for large-scale systems and facilitate the modelling of the simulated agents. Analytical models [43], [44] based on graph and queueing theory have also been used to design and evaluate evacuation schemes. The use of smart techniques to direct the evacuees [45] using adaptive routing algorithms can also alleviate the computational cost and hence algorithmic delay when conditions in the evacuation vary rapidly. In emergencies, the communications infrastructure itself may also be attacked [46] so that back-up communications such as opportunistic networks can be of critical importance [47].

Notation	Definition
r_i	Firing rate for neuron i
$k_i(t)$	Internal state of neuron i at time t
$\Lambda(i)$	Excitatory spike from the outside world to neuron i
$\lambda(i)$	Inhibitory spike from the outside world to neuron i
$p^+(i,j)$	Probability of excitatory spike from neuron i to neuron j
$p^-(i,j)$	Probability of inhibitory spike from neuron i to neuron j
$d(i)$	Probability of departure spike from neuron i to the outside world
$Q(i,j,m)$	Probability of neuron i synchronous interaction together with neuron j to affect third neuron m

TABLE I
List of the RNN training model symbols [23]

The Random Neural Network [24] has been suggested for learning in many engineering problems, such as image processing and video compression [48], route discovery in networks [49] and optimisation at low computational cost [50]. In [23] the RNN is used to solve a NP-hard distributed resource allocation problem approximately in real-time.

III. RNN Based Algorithm for Resource Allocation

Decision-making of the assignment of rescuers to victims must be fast since a longer exposure to hazards will reduce the survival rates of evacuees [51]. Thus we discuss a RNN based fast heuristic to provide near-optimal advice in real-time. The RNN model is first summarised, and we then describe a task assignment algorithm based on the RNN with synchronised interactions [23] to allocate a set of tasks to a limited set of resources quickly and accurately. The notation used for the RNN is shown in Table I, and the known classical result (Theorem 1) [24] is summarised below.

Theorem 1. *Let $\lambda^-(i)$ and $\lambda^+(i)$, $i = 1, ..., N$ be given by the following system of (s*

$$\lambda^-(i) = \lambda(i) + \sum_{j=1}^{N} r_j q_j \left[p^-(j,i) + \sum_{m=1}^{N} Q(j,i,m) \right] \quad (1)$$

$$\lambda^+(i) = \sum_{j=1}^{N} r_j q_j p^+(j,i) + \sum_{j=1}^{N} \sum_{m=1}^{N} q_j q_m r_j Q(j,m,i) + \Lambda(i) \quad (2)$$

where

$$q_i = \lambda^+(i)/(r_i + \lambda^-(i)) \quad (3)$$

If a unique nonnegative solution $\{\lambda^-(i), \lambda^+(i)\}$ exists for the nonlinear system of (s 1 to 3, such that $q_i < 1 \;\; \forall i$ then

$$\pi(\underline{k}) = \prod_{i=1}^{N} (1-q_i) q_i^{k_i} \quad (4)$$

The network will have a solution if $q_i < 1$ and for the inter-neuron interaction paramaters $p^+(i,j)$, $p^-(i,j)$, $d(i)$, and $Q(i,j,m)$ of Table I, we have :

$$d(i) = 1 - \sum_{j=1}^{N} \left[p^+(i,j) + p^-(i,j) + \sum_{m=1}^{N} Q(i,j,m) \right] \quad (5)$$

We use “weights” to represent the rates at which the neurons interact [23], and they are defined as a simple product of the firing rate of a neuron and its corresponding probability probability of interaction with another neuron:

$$w^+(i,j) = r_i p^+(i,j) \tag{6}$$

$$w^-(i,j) = r_i p^-(i,j) \tag{7}$$

$$w(i,j,l) = r_i Q(i,j,l) \tag{8}$$

Combine (s 5 to 8, we can rewrite (5:

$$r_i = \frac{\sum_{j=1}^{N} \left[w^+(i,j) + w^-(i,j) + \sum_{m=1}^{N} w(i,j,m) \right]}{1 - d(i)} \tag{9}$$

The denominator of q_i in (3) can be rewritten with (1), (7), and (8) to get:

$$D(i) = r_i + \lambda(i) + \sum_{j=1}^{N} q_j \left[w^-(j,i) + \sum_{m=1}^{N} w(j,i,m) \right] \tag{10}$$

similarly combining with (2), (6), and (8) we have:

$$N(i) = \sum_{j=1}^{N} q_j w^+(j,i) + \sum_{j=1}^{N} \sum_{m=1}^{N} q_j q_m w(j,m,i) + \Lambda(i) \tag{11}$$

and

$$q_i = \frac{N(i)}{D(i)}. \tag{12}$$

A. Assigning rescuers using the RNN based algorithm

Consider a group of rescuers R waiting at the building exits and some victims V being trapped in the hazardous areas, and assume that the initial locations of both rescuers and injured civilians are known. The approach in [26] adopts an abstract representation which does not explicitly consider the spatial organisation of the area where the emergency is being managed, whereas here we will evaluate the proposed algorithm directly in the context of a three floor building and its specific layout as rooms, corridors, staircases, exits and so on.

In this approach the RNN parameters can be defined as a cost $C(r,v)$ associated with each possible assignment of rescuer r to victim v, a fail execution probability $q(r,v)$ representing that rescuer r is unable to rescue victim v while the fact that the rescuer has been assigned to the victim, and a penalty $K(v)$ for not executing rescue for victim v. In this case, any assignment will have an expected cost and the objective of the algorithm is to minimize the overall expense by selecting the optimal assignments in order to increase the number of injured civilians collected and reduce the total rescue time. The decision of allocating rescuer r to victim v is represented by the probability $p(r,v)$, so that $p(r,v) = 1$ if rescuer r is allocated to victim v, and $p(r,v) = 0$ otherwise. The number of victims and rescuers are $|V|$ and $|R|$, respectively. Using $C(r,v)$, $K(v)$, $q(r,v)$, and $p(r,v)$, the objective function that we minimize is then:

Notation	Definition
$K(v)$	Penalty for not rescuing victim *v*
$q(r,v)$	Probability rescuer *r* is unable to rescue victim *v*
$C(r,v)$	Cost for saving victim *v* by rescuer *r*
$\Lambda(r,v)$	External arrival rate of excitatory signals to neuron *(r, v)*
$\lambda(r,v)$	External arrival rate of inhibitory signals to neuron *(r, v)*
$w^+(r,v;r',v')$	Rate of excitatory signals to neuron *(r, v)* from firing neuron *(r', v')*
$w^-(r,v;r',v')$	Rate of inhibitory signals to neuron *(r, v)* from firing neuron *(r', v')*
$r(r,v)$	Firing rate of neuron *(r, v)*
$Q(r,v)$	Probability neuron *(r, v)* is excited

TABLE II
LIST OF THE RNN PARAMETER ASSOCIATED APPROACH SYMBOLS [26]

$$\begin{aligned} Minimize\ C \quad &= \sum_{v \in V} \sum_{r \in R} C(r,v) p(r,v) \\ &\quad + \sum_{v \in V} K(v) \prod_{r \in R} q(r,v)^{p(r,v)} \end{aligned} \tag{13}$$

$$\begin{aligned} Subject\ to: \quad &(1) \quad p(r,v) \in \{0,1\}, \quad r \in R, v \in V, \\ &(2) \quad \sum_{v \in V} p(r,v) = \pi(r) \in \{0,1\}, \quad r \in R, \end{aligned}$$

where the first term of (13) is the average cost used by the rescuers, and the second term represents the cumulative average cost of failing to rescue the victims. The addition of these two terms indicates that the minimisation of (13 is to find a optimal balance between the successful execution costs and the fail rescue penalties. The first constraint assumes that $p(r,v) = 1$ if rescuer r is allocated to victim v and $p(r,v) = 0$ if not. Since $p(r,v)$ is either 0 or 1, ensures that (a) rescuer r will not be assigned a task if the assignments related to rescuer r increase the overall cost, and (b) a rescuer cannot be re-assigned some other victims if the rescuer already has been assigned a task.

The solution of (13) may be found by enumeration of all possible allocations, but this would be computationally impractical. and the RNN algorithm provides a quick and accurate solution where each possible assignment (r,v) is represented by a neuron $N(r,v)$. The list of symbols used for the approach is summarised in Table II. The excitatory and inhibitory signals' arrival rates of each neuron $N(r,v)$ are specified according to [26],

$$\begin{gathered} \Lambda(r,v) = max\{0, b(r,v)\} \qquad \lambda(r,v) = max\{0, -b(r,v)\} \\ w^-(r,v;r',v) = max\{0, b(r,v)\}, \quad if\ r \neq r' \\ w^-(r,v;r,v') = max\{0, b(r,v)\}, \quad if\ v \neq v' \\ w^+(r,v;r,v') = w^-(r,v;r',v') = 0 \text{ for all other } r,\ r',\ v,\ v' \\ \text{where } b(r,v) = K(v)(1 - q(r,v)) - C(r,v) \end{gathered}$$

For the sake of simplicity, we assume that the synchronous interaction between two neurons cannot affect some third neuron, i.e. $w(i,j,m) = 0$. Hence, we rewrite (9) as:

$$r(r,v) = \sum_{r',v'} w^-(r,v;r',v'), \tag{14}$$

and (12) is then rewritten as:

$$\begin{aligned} Q(r,v) &= \Lambda(r,v)/[\lambda(r,v) + \sum_{r' \neq r} Q(r',v) w^-(r',v;r,v) \\ &\quad + \sum_{v' \neq v} Q(r,v') w^-(r,v';r,v) + r(r,v)] \end{aligned} \tag{15}$$

When some evacuees are trapped in hazardous areas, the rescuers' locations are determined by sensors and communicated to the decision centre by a network. Next, (15) is solved [26] iteratively until convergence to obtain an efficient rescuer dispatch scheme.

Note that the RNN based algorithm represents each rescuer-victim pair as a neuron, so that neuron i associates the rescuer r is allocated with the victim v. The objective function takes into account the cost of allocating rescuer r to victim v, the probability that rescuer r is unable to save victim v, and the penalty for not rescuing v. As a result the algorithm will select a given rescuer-victim pair if the allocation reduces the cost function.

IV. Simulation of the Rescuer Assignment Algorithm

In this section, we begin by presenting the assumptions we make for the simulations, and describe the simulation model, and finally the simulation results are presented. We assume that the health level of evacuees and rescuers decreases gradually if exposed to the hazardous areas, and the evacuees are immobilised if their remaining health is less than a threshold, and need to be evacuated rescuers. In real emergency situations, the rescuers wear protective clothes, and the health level of rescuers decreasesmore slowly than that of civilians. We assume that the maximum capacity of each of them is two, so that each rescuer is allocated one trapped civilian each but allows for the possibility that one more trapped evacuee is assigned if it is on the rescuer's way towards the exits. Finally, rescuers are assumed to be usable only once.

The simulation model is established based on the Distributed Building Evacuation Simulator [28], where all simulated entities including evacuees as well as rescuers are modelled as agents and these entities are able to communicate with each other. We assume that this is a fire-related emergency evacuation and the area to evacuate is a building based on the three lower floors of Imperial College London's EEE building [45]. A graph representation of this building is showed in Figure 1.

The evacuees try to move to any of the two exits along a safe path to avoid injury. But in case of danger they send a message that includes their identity and location, and an indication of their health status to ask help from the rescuers. On the other hand, the objective of the rescuers is to reach the potential victims who are in dangerous areas and move or carry them to safe sites and exits. tWhen the evacuees need help, the RNN based algorithm determines the rescuer to victim assignments that sends a set of rescuers to the evacuees that have requested for help. To move towards the evacuees the rescuers use Dijkstra's shortest path algorithm since are assumed to be able to protect themselves from hazards such as fire. The rescuers move the injured evacuees using the Dijkstra's algorithm where the hazard intensity and the physical distance together are being minimised [52].

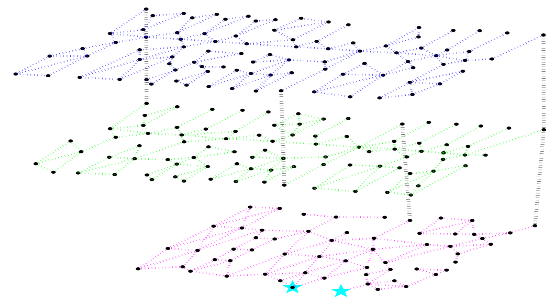

Fig. 1. Simplified interior space plan of the building model that is used in the simulations [45]. The two exits on the ground floor are marked by two stars.

The effectiveness of the proposed RNN based task assignment algorithm was evaluated in terms of two performance metrics:

- (1) The percentage of evacuees who are saved by the rescuers based with respect to the total number of injured civilians, as a function of the total number of evacuees in the system;
- (2) The number of victims saved by the rescuers with the size of the set of rescuers and the set of victims that have requested help.

The results of our experiments are obtained by comparing two different scenarios, without, and with the RNN based task assignment algorithm. Different numbers of rescuers and of victims are considered.

Each simulation result is obtained by executing 10 simulation runs. For each simulation run, the locations of the injured civilians are randomised around the hazards in the building. The purpose of the simulations is to check if the RNN based algorithm can avoid wasting resources. Thus, to keep things as simple as possible, the cost of each assignment $C(r,v)$ and the probability that a rescue fails $q(r,v)$ are taken to be independent of the victim v [26] and are generated randomly.

$K(v)$, $C(r)$, and $q(r)$ are generated from a uniform distribution in the interval $[0, 50]$, $[0, 10]$, and $[0.05, 0.15]$ respectively, and $Q(r,v)$ is initialised to zero for all assignments. All RNN parameters are updated 20 times for each allocation to reach convergence of the RNN state, and the assignment with the highest probability $Q(r,v)$is selected for the decision. The results are summarised in Figure 2 and Table III.

In Figure 2, it is clear that when the RNN algorithm is *not used* the percentage of rescued civilians can *decrease* with the *increase of the number of evacuees in the building*, and this is due to the fact that it takes a relatively long time to evacuate

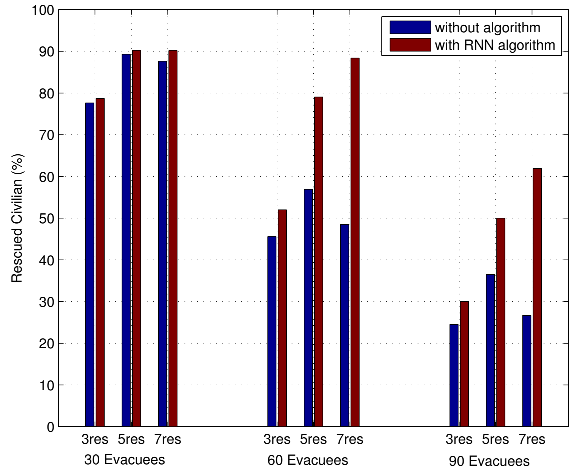

Fig. 2. The percentage of people who require rescuing and make it out safely from the building as a function of the number of evacuees.

No. of rescuers $\lvert R\rvert$	No. of victims $\lvert V\rvert$	Without algorithm	With the RNN algorithm
3	4	3.00	3.80
5	4	3.43	4.00
7	4	4.00	4.00
3	8	4.71	5.75
5	8	6.00	7.50
7	8	5.89	8.00
3	16	4.88	6.00
5	16	7.40	9.75
7	16	6.67	13.25

TABLE III
NUMBER OF EVACUEES SAVED BY THE RESCUERS

people in high population density environments because of congestion, which also increases the time of exposure of evacuees to hazards and reduces the number of survivors. Surprisingly, ithout the task assignment algorithm, the system with 7 rescuers has a lower percentage of rescued civilians compared with the system with 5 rescuers for each case (30, 60, and 90 evacuees). This is even more obvious with high population densities (60 and 90 evacuees). The reason is that when there are people who need help, the rescuers enter the building and move towards the areas of greatest concern, so that some increase in congestion can occur [34]. Thus, in such circumstances both the rescuers and the evacuees may spend more time in unsafe parts of the building.

On the other hand, without the task assignment algorithm (1) different rescuers may be allocated to the same victim, (2) different victims may be assigned to the same rescuer. With the RNN based algorithm, the percentage of rescued civilians tends to increase with the number of rescuers in the system as shown in Figure 2, and as would expected. Although having more rescuers can also increase congestion, the RNN based algorithm does save more people and performs relatively better with higher numbers of people.

In Table III, given a set of victims, the increase of the number of rescuers cannot significantly improve the performance on the number of rescued evacuees without the task assignment algorithm and more rescuers may reduce the number of rescued evacuees, which agrees with the results obtained from Figure 2. Furthermore, the RNN based algorithm can provide a near-optimal performance, where

$$
\begin{aligned}
&If\ \ |V| < overall\ maximum\ capacity\ of\ rescuers\\
&\qquad number\ of\ rescued\ evacuees \xrightarrow{approx} |V|\\
&else\\
&\qquad number\ of\ rescued\ evacuees\\
&\xrightarrow{approx} overall\ maximum\ capacity\ of\ rescuers\\
&end.
\end{aligned}
$$

V. FUTURE WORK

We have evaluated a rescuer assignment algorithm to efficiently dispatch rescuers to aid victims in an emergency. The simulations indicate that near-optimum rescuer allocation can significantly improve the outcome. In future work, the task assignment algorithm can be re-designed when the identity of the rescuers and the identity and nature of the potential victims is known and is part of the costs, leading to a "multiclass" formulation [24] of the approach. When there are fewer rescuers than injured civilians, rescuers could be used to find victims [53], then cluster them [54] according to their locations, and rescuers may then evacuate several at the same time. To reduce congestion and queuing of both rescuers and victims [55], [56], smart routing schemes [45] may be used to offer faster evacuation and avoid the congestion that occurs along the safest or shortest paths that all the evacuees may try to use simultaneously.

REFERENCES

[1] Q. Li, M. D. Rosa, and D. Rus, "Distributed algorithms for guiding navigation across a sensor network," in *ACM Int'l Conf. Mobile Computing and Networking*, 2003, pp. 313–325.

[2] D. J. Malan, T. R. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, and S. Moulton, "Sensor networks for emergency response: Challenges and opportunities," *IEEE Pervasive Computing*, vol. 3, no. 4, pp. 16–23, 2004.

[3] C. Fischer and H. Gellersen, "Location and navigation support for emergency responders: A survey," *IEEE Pervasive Computing*, vol. 9, no. 1, pp. 38–47, 2009.

[4] E. Gelenbe and F.-J. Wu, "Large scale simulation for human evacuation and rescue," *Computers and Mathematics with Applications*, vol. 64, no. 12, pp. 3869–3880, Dec. 2012.

[5] M.-S. Pan, C.-H. Tsai, and Y.-C. Tseng, "Emergency guiding and monitoring applications in indoor 3D environments by wireless sensor networks," *Int'l J. Sensor Networks*, vol. 1, no. 1/2, pp. 2–10, Sept. 2006.

[6] Y.-C. Tseng, M.-S. Pan, and Y.-Y. Tsai, "Wireless sensor networks for emergency navigation," *IEEE Computer*, vol. 39, no. 7, pp. 55–62, Jul. 2006.

[7] E. Gelenbe and Y. Cao, "Autonomous search for mines," vol. 108, no. 2, pp. 319–333, 1998.

[8] E. Gelenbe and T. Koçak, "Area-based results for mine detection," *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 38, no. 1, pp. 12–24, 2000.

[9] ——, "Area based results for mine detection," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, pp. 12–24, 2000.

[10] S. H. Bokhari, *Assignment Problems in Parallel and Distributed Computing*. Kluwer Academic, 1987.

[11] D. W. Pentico, "Assignment problems: A golden anniversary survey," *European Journal of Operational Research*, vol. 176, pp. 774–793, Jan. 2007.

[12] J. Aguilar and E. Gelenbe, "Task assignment and transaction clustering heuristics for distributed systems," *Information Sciences*, vol. 97, no. 1-2, pp. 199–219, Mar. 1997.

[13] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Courier Dover Publications, 1998.

[14] B. George, S. Shekhar, and S. Kim, "Spatio-temporal network databases and routing algorithms," *Transactions on Knowledge and Data Engineering (TKDE), IEEE,(Submitted in 2008, Also Tech. Report 08-039, Computer Sc., Univ. of Minnesota)*, 2008.

[15] E. Gelenbe, "Sensible decisions based on qos," *Computational management science*, vol. 1, no. 1, pp. 1–14, 2003.

[16] Q. Lu, Y. Huang, and S. Shekhar, "Evacuation planning: A capacity constrained routing approach," in *Intelligence and Security Informatics*. Springer, 2003, pp. 111–125.

[17] Q. Lu, B. George, and S. Shekhar, "Capacity constrained routing algorithms for evacuation planning: A summary of results," in *Advances in spatial and temporal databases*. Springer, 2005, pp. 291–307.

[18] T. Tabirca, K. N. Brown, and C. J. Sreenan, "A dynamic model for fire emergency evacuation based on wireless sensor networks," *Parallel and Distributed Computing*, pp. 29–36, 2009.

[19] Y. Bo, W. Yong-gang, and W. Cheng, "A multi-agent and GIS based simulation for emergency evacuation in park and public square," in *Int'l Conf. Computational Intelligence and Security*, 2009, pp. 211–216.

[20] F. ud Din Ghauri, S. ur Rehman, M. Yasir, and S. Asghar, "Multi agent based decision support system for prioritized emergency fire evacuation," in *Int'l Conf. New Trends in Information Science and Service Science*, 2010, pp. 457–463.

[21] Y. Nakajima, H. Shiina, S. Yamane, and T. Ishida, "Disaster evacuation guide: Using a massively multiagent server and GPS mobile phones," in *IEEE Int'l Symp. Applications and the Internet*, 2007.

[22] L. Han, S. Potter, G. Beckett, G. Pringle, S. Welch, S.-H. Koo, G. Wickler, A. Usmani, J. L. Torero, and A. Tate, "FireGrid: An e-infrastructure for next-generation emergency response support," *Journal of Parallel and Distributed Computing*, vol. 70, no. 11, pp. 1128–1141, 2010.

[23] E. Gelenbe and S. Timotheou, "Random neural networks with synchronised interactions," *Neural Computation*, vol. 20, no. 9, pp. 2308–2324, Jul. 2008.

[24] E. Gelenbe and J.-M. Fourneau, "Random neural networks with multiple classes of signals," *Neural Computation*, vol. 11, no. 4, pp. 953–963, May 1999.

[25] E. Gelenbe, "The first decade of g-networks," *EUROPEAN JOURNAL OF OPERATIONAL RESEARCH*, vol. 126, no. 2, pp. 231–232, October 2000.

[26] E. Gelenbe, S. Timotheou, and D. Nicholson, "Fast distributed near-optimum assignment of assets to tasks," *The Computer Journal*, vol. 53, no. 9, pp. 1360–1369, 2010.

[27] E. Gelenbe, S. Timotheou, and D. Nicholson, "A random neural network approach to an assets to tasks assignment problem," in *SPIE Symp. Defence, Security and Sensing, Track of Signal Processing, Sensor Fusion, and Target Recognition XIX*, 2010, p. 76970Q.

[28] N. Dimakis, A. Filippoupolitis, and E. Gelenbe, "Distributed building evacuation simulator for smart emergency management," *The Computer Journal*, vol. 53, no. 9, pp. 1384–1400, Feb. 2010.

[29] J. Reich and E. Sklar, "Robot-sensor networks for search and rescue," in *IEEE Int'l Workshop Safety, Security and Rescue Robotics*, 2006.

[30] E. Gelenbe and F.-J. Wu, "Future research on cyber-physical emergency management systems," *Future Internet*, vol. 5, no. 3, pp. 336–354, 2013.

[31] A. Verma, H. Sawant, and J. Tan, "Selection and navigation of mobile sensor nodes using a sensor network," *Pervasive and Mobile Computing*, vol. 2, no. 1, pp. 65–84, 2006.

[32] I. Stojmenovic, D. Liu, and X. Jia., "A scalable quorum based location service in ad hoc and sensor networks," *Int'l J. Comm. Networks and Distributed Systems*, vol. 1, no. 1, pp. 71–94, 2008.

[33] X. Li, N. Santoro, and I. Stojmenovic, "Localized distance-sensitive service discovery in wireless sensor and actor networks," *IEEE Trans. Computers*, vol. 58, no. 9, pp. 1275–1288, Sept. 2009.

[34] S. Li, A. Zhan, X. Wu, and G. Chen, "Ern: Emergence rescue navigation with wireless sensor networks," *Parallel and Distributed Systems*, pp. 361–368, 2009.

[35] D. Chen, C. K. Mohan, K. G. Mehrotra, and P. K. Varshney, "Distributed in-network path planning for sensor network navigation in dynamic hazardous environments," *Wireless Comm. and Mobile Computing*, 2010.

[36] M. Li, Y. Liu, J. Wang, and Z. Yang, "Sensor network navigation without locations," in *IEEE INFOCOM*, 2009, pp. 2419–2427.

[37] P.-Y. Chen, W.-T. Chen, and Y.-T. Shen, "A distributed area-based guiding navigation protocol for wireless sensor networks," in *IEEE Int'l Conf. Parallel and Distributed Systems*, 2008, pp. 647–654.

[38] X.-Y. Li, G. Calinescu, P.-J. Wan, and Y. Wang, "Localized Delaunay triangulation with application in ad hoc wireless networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 14, no. 10, pp. 1035–1047, 2003.

[39] J. Bruck, J. Gao, and A. A. Jiang, "MAP: medial axis based geometric routing in sensor networks," in *ACM Int'l Conf. Mobile Computing and Networking*, 2005, pp. 88–102.

[40] M. Barnes, H. Leather, and D. K. Arvind, "Emergency evacuation using wireless sensor networks," in *IEEE Conf. Local Computer Networks*, 2007, pp. 851–857.

[41] A. Cherniak and V. Zadorozhny, "Towards adaptive sensor data management for distributed fire evacuation infrastructure," in *Int'l Conf. Mobile Data Management*, 2010, pp. 151–156.

[42] G. Xing, T. Wang, Z. Xie, and W. Jia, "Rendezvous planning in wireless sensor networks with mobile elements," *IEEE Trans. Mobile Computing*, vol. 7, no. 12, pp. 1430–1443, 2008.

[43] J. Smith, "State-dependent queueing models in emergency evacuation networks," *Transportation Research Part B: Methodology*, vol. 25, no. 6, pp. 373–389, Feb. 1991.

[44] A. Desmet and E. Gelenbe, "Graph and analytical models for emergency evacuation," *Future Internet*, vol. 5, no. 1, pp. 46–55, Feb. 2013. [Online]. Available: http://www.mdpi.com/1999-5903/5/1/46

[45] H. Bi, A. Desmet, and E. Gelenbe, "Routing emergency evacuees with cognitive packet networks," in *Proceedings 28th Annual International Symposium on Computer and Information Sciences ISCIS 2013*. Springer, 2013.

[46] E. Gelenbe and G. Loukas, "A self-aware approach to denial of service defence," *Computer Networks*, vol. 51, no. 5, pp. 1299–1314, 2007.

[47] G. Görbil and E. Gelenbe, "Opportunistic communications for emergency support systems," *Procedia Computer Science*, vol. 5, pp. 39–47, 2011.

[48] E. Gelenbe, M. Sungur, C. Cramer, and P. Gelenbe, "Traffic and video quality with adaptive neural compression," *Multimedia Systems*, vol. 4, no. 6, pp. 357–369, Dec. 1996.

[49] E. Gelenbe, G. Sakellari, and M. d'Arienzo, "Admission of qos aware users in a smart network," *ACM Transactions on Autonomous and Adaptive Systems*, vol. 3, no. 1, pp. 14–28, Mar. 2008.

[50] E. Gelenbe, V. Koubi, and F. Pekergin, "Dynamical random neural network approach to the travelling salesman problem," *Proc. IEEE Symp. Syst., Man, Cybern.*, vol. 2, pp. 630–635, Oct. 1993.

[51] S. Li, A. Zhan, X. Wu, P. Yang, and G. Chen, "Efficient emergency rescue navigation with wireless sensor networks," *Journal of Information Science and Engineering*, vol. 27, no. 1, pp. 51–64, 2011.

[52] A. Filippoupolitis and E. Gelenbe, "A distributed decision support system for building evacuation," in *Human System Interactions, 2009. HSI'09. 2nd Conference on*, 2009, pp. 323–330.

[53] E. Gelenbe, N. Schmajuk, J. Staddon, and J. Reif, "Autonomous search by robots and animals: A survey," *Robotics and Autonomous Systems*, vol. 22, no. 1, pp. 23 – 34, 1997.

[54] Y.-C. Wang, W.-C. Peng, M.-H. Chang, and Y.-C. Tseng, "Exploring load-balance to dispatch mobile sensors in wireless sensor networks," *IEEE Int'l Conf. Computer Comm. and Networks*, pp. 669–674, Aug. 2007.

[55] E. Gelenbe and R. R. Muntz, "Probabilistic models of computer systems. part I (exact results)," *Acta Informatica*, vol. 7, pp. 35–60, 1976.

[56] E. Gelenbe, "Probabilistic models of computer systems. part II: Diffusion approximations, waiting times and batch arrivals," *Acta Informatica*, vol. 12, pp. 285–303, 1979.