A Decision Support System for Disaster Management in Buildings

Avgoustinos Filippoupolitis and Erol Gelenbe
Department of Electrical & Electronic Engineering,
Imperial College London,
SW7 2BT, UK
{afil, e.gelenbe}@imperial.ac.uk

Keywords: Decision Support Systems, Emergency Response, Building Evacuation

Abstract

Information systems that provide decision support in an efficient and timely manner can prove beneficial for emergency response operations. In this paper we propose the use of a system that provides movement decision support to evacuees by directing them through the shortest or less hazardous routes to the exit and evaluate it with a specialised software platform that we have developed for simulation of disasters in buildings. The system operates in a distributed manner, and computes the best evacuation routes in real-time while a hazard is spreading inside the building. It is composed of a network of decision nodes and sensor nodes, positioned in specific locations inside the building. The recommendations of the decision nodes are computed in a distributed manner, at each of the decision nodes, which then communicate them to evacuees or rescue personnel located in their vicinity. We use a multi-agent simulation platform for Building Evacuation that we developed, in order to evaluate our proposed system in various emergency scenarios. Our simulation results show that the overall outcome of the evacuation procedure is improved when the decision support system is in operation.

1. INTRODUCTION

Fast decision making is indispensable during an emergency situation, since it directly affects the overall outcome of an evacuation procedure. The total evacuation time, the exposure level of the evacuees to the hazard and the injuries they sustain are closely related to the time spent in deciding which direction to follow while moving inside a building. Correctly asserting the situation and deciding regarding the best path to follow is not, however, an easy task for an evacuee. The best path towards an exit is often not known, unless we assume everyone has a thorough knowledge of the building's structure. Finding a safe path towards an exit becomes even more difficult, if we take into account the constantly changing conditions due to the spreading of a hazard [1] (e.g. a fire or a hazardous gas) in various locations. The evacuees could benefit from a decision support system which operates inside the building and provides them with directions regarding the best available exit. By following its directions they can safely exit the building, avoiding paths that could expose them to the hazard.

Our approach aims at designing and evaluating a system which can be used during an emergency situation inside a building. The system must be able to function in real-time, adapt to the changes of the environment and provide reliable suggestions to the evacuees regarding the direction of the best available exit. This approach has been inspired by [2], where vehicles, modelled as smart agents, are traversing a dangerous urban grid. The vehicles use information coming from the environment and from other vehicles and are able to adapt in order to travel rapidly and safely. In our case, the environment consists of a building where civilians are taking part in an evacuation in the presence of a spreading hazard. The evacuees follow the directions provided by the decision support system and are able to exit the building using the best available paths, while avoiding the hazardous areas. We evaluate the prosposed system using a multi-agent Building Evacuation Simulator which models the evacuation procedure. This provides a realistic environment in which our proposed method can be tested.

The rest of the paper is organised as follows: Section 2. presents some of the relative literature on the subject. Section 3. gives the details of the modelling approach we followed for the design of the system. In Section 4. we present the distributed algorithm that is used by the decision support system. The performance evaluation of the system and the simulation results for various scenarios are the subjects of Section 5. and 6. respectively. We conclude with a summary of our contributions and possible extensions to our work.

2. RELATED WORK

The problem of finding the best path to a specified location in order to guide humans towards an exit or an area of interest, has been approached in various ways.

An algorithm that aims at navigating a robot using a predeployed sensor network is proposed in [3]. The authors use the value iteration algorithm for determining the direction towards which the robot should move, while the relevant calculations are performed by the sensor network in a distributed manner. They evaluate their approach using a wireless sensor network of nine nodes and a robot that has to navigate from a "home" node to a destination. The robot is able to navigate inside the network, but the approach is not tested under dynamic conditions since the cost of the links remains static. Moreover, parameters inherent to human navigation during a disaster, such as potential congestion or hazard along paths, are not taken into account.

In [4] the authors use a sensor network to navigate a flying robot. The path calculation algorithm is based on a routing protocol for sensor networks. The robot gathers lists of path segments from multiple sensors as it moves and is able to assemble the entire path. A network of 54 wireless sensor nodes was used for the experimental evaluation of the approach. The nodes where positioned in a grid topology while a robot helicopter, equipped with a wireless sensor, hovered over them. The helicopter was able to successfully follow the paths suggested to it by the sensor network. The proposed system was extended for guiding humans, but the approach consisted of only one human and twelve sensors positioned inside a building. The approach was not evaluated for larger building occupancies and dynamic conditions such as the spreading of a hazard.

In [5] the authors propose a system that uses wireless sensor nodes in order to navigate a robot or a human towards an exit. The system calculated the best path towards the goal location, by avoiding the hazardous areas. There is no evaluation, however, for the case of a dynamic hazard that spreads in different locations. Moreover, the system is not tested in a scenario that includes a large number of evacuees.

Finally, a similar system is prosposed in [6]. The authors propose the use of a sensor network in order to calculate a path that leads to an exit and does not pass through the hazardous area. The simulation results demonstrate the ability of the algorithm to find the safest paths. However, the simulations do not model a hazard spreading inside the network or the movement of a humans that try to find the best path towards the exit.

3. THE DECISION SUPPORT SYSTEM

The design approach we adopted is based on assumptions related to the nature of our problem and the environment in which our system operates. We first assume that the layout of the building is known. A decision support system should be pre-installed in the building before the evacuation process is initiated. It is thus valid to consider the we have full knowledge of the building's structure, as a system like this has already been deployed when the emergency situation occurs. We have also assumed that a number of Decision Nodes(DNs) are installed in specific locations inside the building. The DNs are not required to have high processing power or storage. Their role is to compute the direction that should be followed by each evacuee, towards the best available exit. The suggestion of a DN is communicated to people in its vicinity via a visual indicator (such as a smart panel) or a wireless commu-

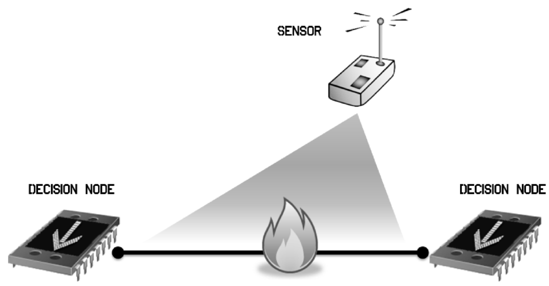

Figure 1. Sensor node monitoring hazard on a link between two decision nodes. The arrows on the panels point to the direction towards the best available exit

nication device (i.e. a PDA) which is carried by the evacuees or the emergency personnel and receives the suggestions from the DNs. Our last assumption is the existence of a network of sensor nodes, which provide the DNs with real-time information related to the conditions inside the building (such as high temperature or smoke due to a the presence of a fire).

Using the known building layout, we construct a graph G. The vertices of the graph correspond to locations where people can congregate, such as rooms, corridors and doorways. A link between two vertices of the graph represents a path that can be followed by the evacuees. The length $l(i, j)$ of the link between two vertices of the graph, represents its physical distance. Each sensor is associated with each link (i, j) and monitors its hazard intensity $H(i, j)$. Under normal conditions (when there is no hazard present) $H(i, j) = 1$. The value of $H(i, j)$ will increase as the value of the monitored hazard increases.

We can now introduce a metric that combines the length $l(i, j)$ and the hazard $H(i, j)$ along a link. It expresses how hazardous a link is for a civilian that will traverse it. Thus, we define the *effective length* $L(i, j)$ of a link as: $L(i, j) = l(i, j) \cdot H(i, j)$.

The value of the effective length depends both on the physical length of a link and on the value of the hazard along that link. When there is no hazard present, $L \equiv l$, since the effective length is equivalent to the physical length of the link. The higher the value of L for a link, the more hazardous it becomes for a civilian to move along it. The DNs and the sensor nodes are positioned in specific locations inside a building. Each DN is placed at each of the vertices of the graph G. In practice, however, there could be fewer decision nodes than vertices in G, with each DN being in charge of providing decisions for a set of contiguous locations of G. Figure 1 depicts a sensor node that monitors the hazard value on a link along two decision nodes.

4. A DISTRIBUTED ALGORITHM

Instead of using a centralised system to compute the value of the effective length of the paths to an exit, we propose a distributed architecture. A centralised implementation of the proposed system would involve a central processor responsible for gathering information from the sensor nodes, executing the algorithm and communicating the resulting decisions to the decision nodes. However, this approach is not fault tolerant, since it heavily relies on a single entity (i.e. the processing centre). In the case of a failure, the decision system cannot function. Our approach uses local communications between neighbouring DNs, which is more robust compared to a centralised solution which relies on being always able to communicate with the processing centre. Moreover, it is quite possible that a single processing centre cannot meat the computational requirements of a large scale system. As the size of a building increases, the required number of DNs increases accordingly, and this has a direct impact on the amount of information that has to be processed by the centre. In our distributed system, each DN executes the decision support algorithm locally and relies only on its own computational resources. Thus, the size of the system does not affect the size of data that have to be processed. Finally, a distributed system guarantees an extensible design process, since the design principles for a given size of building can be scaled up or down, to address larger or smaller buildings.

The algorithm that we propose is inspired by the distributed shortest path problem [7, 8, 9] and from adaptive routing techniques such as Cognitive Packet Networks [10]. Each Decision Node executes the algorithm, which calculates the next Decision Node that is on the best available path towards an exit.

A Decision Node, at vertex u, stores the following information:

- The effective length L of all the links that are incident to u
- For every neighbour n of u, the effective length of the path from n to an exit $e : L(n,e)$
- The effective length of the shortest path from u to an exit e: $L(u,e)$
- The next suggested Decision Node

The initial value for $L(u,e)$ is set to zero if node u is an exit, otherwise it is set to infinity. We can however consider that the initial condition for each Decision Node,at that time, is the actual physical length $l(u,e)$ of the shortest path from a decision node u to an exit e. This is a consequence of the fact that the system will be already operating before the hazard occurs, thus each Decision Node will have selected a path that minimises the effective length when no hazard is present. We should also point out that it is not necessary for a Decision Node to keep information regarding the effective length L of the paths towards all the available exits. As the algorithm is executed, this information is propagated from all the exits to all the Decision Nodes. Each Decision Node will eventually select the exit that minimises the value of the selected metric, which in our case is the effective length of the path from the node to the exit. The selection of an exit depends on the location of the Decision Node, the locations of the exits and the spreading of the hazard. We must finally note that the DNs do not store the complete path towards an exit. Each DN only stores the identity of the next suggested DN which lies along the best path leading to a building exit.

Each DN, at vertex u, periodically executes the following steps of the algorithm:

1. **Send** to every neighbour n of u, the effective length of the path from u to the exit $e : L(u,e)$
2. **Request** the hazard intensity H from each sensor node that monitors a link incident to u
3. **Calculate** the effective lengths $L(u,n)$,where n is a neighbour of u
4. **Update** the effective length $L(u,e)$ of the shortest path x to the exit:
$L(u,e)$ = min $\{L(u,n)+L(n,e)$: $\forall$ neighbours n of $u\}$
5. **Sets** the next suggested DN v:
v = argmin $\{L(u,n) + L(n,e)$: $\forall$ neighbours n of $u\}$

The output of the algorithm is a suggestion towards the evacuees that are located near a DN. The suggestion is of the form **"go to v"**, where v is one of the neighbour decision nodes of u. As the conditions inside the building will change rapidly due to the spreading of the hazard, the DNs will periodically execute the algorithm, update the distance information and communicate the most recent valid advice to the evacuees.

5. PERFORMANCE EVALUATION OF THE SYSTEM

We have implemented the proposed decision support system inside the Building Evacuation Simulator (BES) [11, 12], in order to evaluate its performance during an emergency situation inside a building. The BES is an agent-based simulator for building evacuation. The actors that take part in the simulation are modeled as agents. Each one has its own health model and movement model that depend on the individual characteristics of the actors and on the environment. The implementation of the agents was done using the JADE framework [13]. The physical world of a building is viewed as a collection of "Points of Interest" (PoI) and available links between them, which form a directed graph. The links represent the walking distance between adjacent PoI. The BES

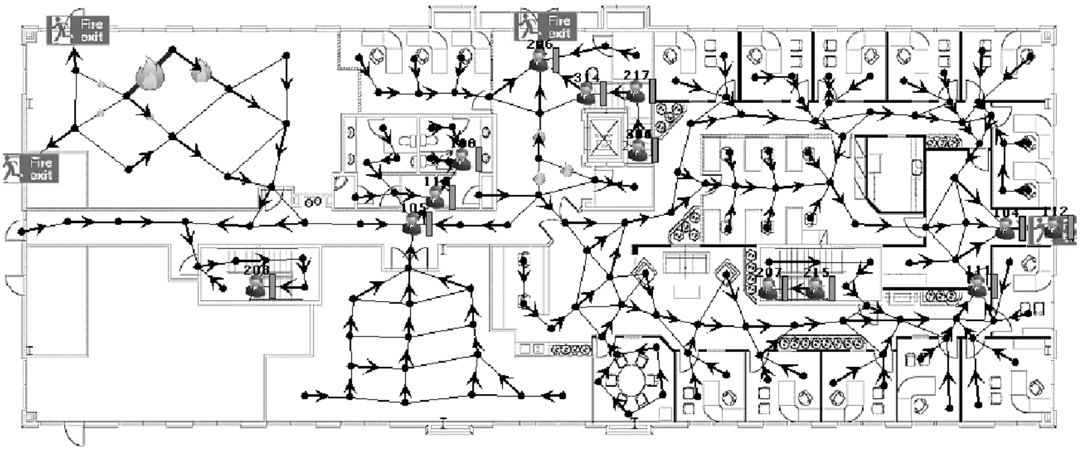

Figure 2. The decision support system operating inside the BES. The suggestions of the DNs are indicated by the arrows

is also able to simulate congestion and the spread of a hazard inside a building [14] , such as a fire or hazardous gas. The BES is composed of actors that move and interact inside the simulated world, while the environment is changing due to the spreading hazard. Different types of actors, such as civilians, rescuers and firefighters can participate in the simulation while the decision support system is used to provide them with directions regarding the best available exit route. Thus, we can investigate the efficiency of our system under realistic conditions and evaluate its performance by analysing the outcome of numerous simulated scenarios. Since our proposed system is based on a graph, which is similar to the one used by the BES, we were able to develop the decision model without the need for major modifications in the organisation of the simulator, which would have resulted in unnecessary complexity and difficulty. Finally, the modular, agent-based structure of the BES allowed for a relatively straightforward addition of the entities that are part of the decision support system, such as Decision Nodes and sensors.

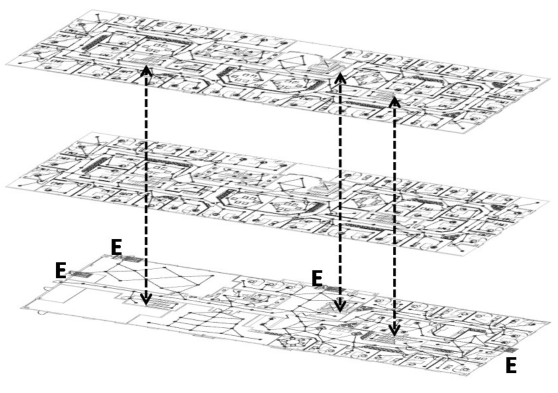

Figure 3. The building used for the evaluation of the decision support system

The evacuation scenarios in which we test the decision support system, take place inside the three storey building depicted in Figure 3. There are three stair cases inside the building, which provide access to the different floors, while a fire starts spreading in multiple locations inside the ground floor. When we use the decision support system during the evacuation procedure, the civilians move according to the directions of the DNs. Figure 2 depicts the ground floor of the building through the BES graphical interface, where the fire has spread in different locations inside the floor. Each arrow represents the recommendation of the respective DN. The DNs are able to communicate these recommendations to the evacuees by either a visual indicator (such as a smart panel) or a wireless communication device (such as a PDA) which is carried by the evacuees. Each DN directs the evacuees towards the best of its neighbouring DNs, along the safest path that leads to one of the building's exits. When the evacuation procedure takes place without the use of the decision support system, we have assumed that the evacuees have perfect knowledge of the building's structure before the fire starts spreading. More specifically, we consider that each evacuee is familiar with all the available exits and is able to follow the shortest path towards one of them. This effectively means that each evacuee has full knowledge of the building's graph and decides his next destination by the use of Dijkstra's algorithm. When an evacuee reaches a hazardous area, he updates his representation of the building's graph and recalculates the shortest path.

5.1. Configuration of the Experiments

Between successive simulation runs, we have chosen to randomise a number of scenario parameters, so that we could test our decision support system in various conditions. The civilians' initial locations on each of the floors, are chosen from a uniform distribution over the respective graph vertices. This randomisation helps us evaluate the performance of the system under different building occupancy patterns. Moreover, the spreading rate of the hazard is different between consecutive simulation runs, since it is based on the probabilistic hazard model used by the Building Evacuation Simulator [11, 14]. This allows us to test the effectiveness of the system under various hazard spreading speeds.

The outcome of the evacuation procedure, which reflects the effectiveness of our decision support system, is evaluated by the following set of metrics:

1. *Percentage of evacuees that have exited the building versus the evacuation time*. This metric measures how fast the civilians are directed towards the building exits

2. *Percentage of evacuees that have sustained fatal injuries versus the evacuation time*. An evacuee that follows the directions of the decision support system, is expected to avoid contact with the hazard. This metric allows us to determine for how long the evacuees where able to remain safe inside the building until they where fatally injured due to extensive exposure to the hazard

3. *Average remaining health of the evacuees*. It denotes the degree to which an evacuee is exposed to the hazard. A low value indicates that the system has succeeded in directing the evacuees along safe paths

4. *Percentage of fatally injured evacuees*. It measures the number of evacuees that were not able to exit the building due to excessive exposure to the hazard.

We have evaluated our system for various values of the algorithm execution frequency . We have also included a case where there is no decision support system in the building during the evacuation procedure and compared the performance of our system against it. For each of these cases, we executed two hundred simulation runs.

6. SIMULATION RESULTS

In this section we present the simulation results from various evacuation scenarios and comment on the effectiveness of the decision support system.

6.1. Low building occupancy

The first simulation scenario investigates the case were there are 30 civilians located inside the building, ten in each of the building's floors. In Figure 4(a) we can see the percentage of evacuees that have excited the building , versus the evacuation time. We note that when the decision support system is used, the evacuation procedure finishes faster and with a higher percentage of civilians exiting the building. Figure 4(b) allows us to see that when the decision support system is used, the civilians are given directions that lead them away from hazardous areas for a longer time period. In other words, they are able to avoid contact with the hazard for as long as possible. This aspect of the system's behaviour can prove very useful in an operation where emergency personnel enter the building in order to find trapped civilians and lead them to the exit. The average remaining evacuee health shown in Figure 4(c), is also higher when the system is used. This result confirms that the use of the system helps the evacuees avoid exposure to the hazard. Finally, Figure 4(d) illustrates the percentage of fatally injured evacuees. We can again verify that the decision support system improves the outcome of the evacuation procedure.

6.2. High building occupancy

In this simulation scenario 20 civilians are located on each floor of the building shown in Figure 3, and try to reach one of the exits located in the ground floor. Figure 5(a) illustrates the percentage of evacuees that have excited the building, versus the evacuation time. We can note that the outcome of the cases where the decision system is present is better compared to the case where the system is absent. We can verify this by comparing the slope and the height of the respective curves. In Figure 5(b) we can see that the system manages to keep the civilians away from the hazard for a longer period of time, compared to the case where the system is not used. This can be verified by the larger size of the time window ending at the time instant of the last fatally injured evacuee. Figure 5(c) depicts the average remaining health of the evacuees. We can verify that the decision support system directs the evacuees away from hazardous areas and towards the best available exit. Figure 5(d) shows the percentage of fatally injured evacuees. We can note that the use of the decision support system results in minimal casualties during the emergency situation , providing superior results compared to the case where the system is absent.

6.3. Comments on the Simulation Results

By comparing the results of the two simulation scenarios, we can observe that the higher occupancy level of the building affects the value of the evacuation time. This is due to the fact that civilians need more time to exit the building because of the increased congestion. However, as it can be verified by Figure 5(a), the use of the decision support system still results in a faster and safer evacuation. We must also note that when the occupancy level of the building increases, the absence of the decision support system results in an even lower average remaining health, as shown in Figure 5(c). This is explained by the fact that the congestion level is now higher and the evacuees need more time to move inside the building. Thus, the hazard has time to spread in more areas and the probability of being exposed to it increases.

Another important parameter that affects the system's performance, is the execution frequency of the distributed algo-

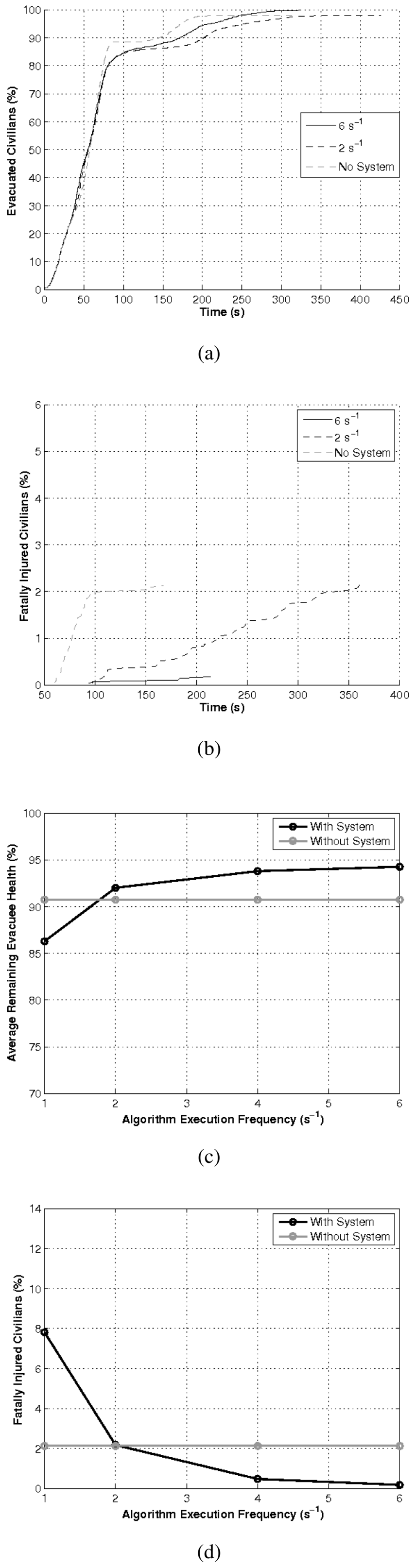

Figure 4. Simulation results for the first evacuation scenario

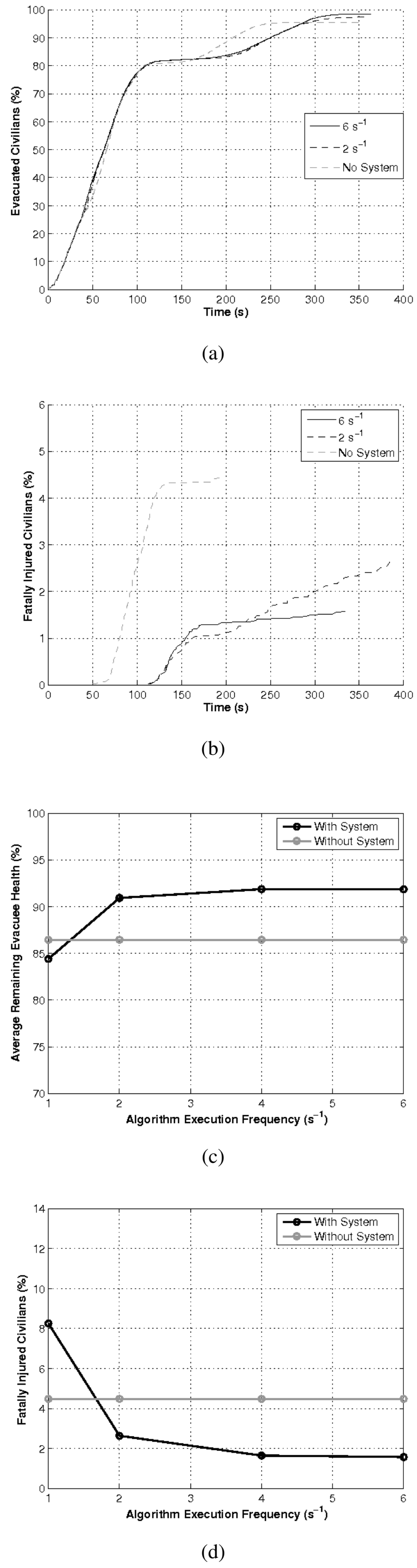

Figure 5. Simulation results for the second evacuation scenario

rithm. This can be clearly seen in Figures 4(c)-(d) and Figure 5(c)-(d). We can explain this aspect of the system's behaviour bye taking into consideration that the propagation of the changes which take place in the dynamic building environment (such as the change on the effective length value along a link), depend on the execution frequency. A high algorithm execution frequency results in a more adaptive system which is able to give faster, more accurate suggestions to the evacuees. This however results in a higher number of message transmissions. On the contrary, lower execution frequencies result in inferior performance. In Figures 4(a) and 5(a), where the percentage of evacuated civilians decreases along with the algorithm execution frequency, we can notice this phenomenon. We should also note that for the lowest value of the execution frequency($1\ s^{-1}$) , the performance of the system is significantly degraded. This is due to the fact that the changes in the links' effective length cannot propagate in time and the DNs are not able to successfully calculate the best path towards an exit.

7. SUMMARY

In this paper we presented a distributed decision support system designed for providing directions to evacuees during the evacuation of a building in the presence of a spreading hazard. The system is composed of a number of decision nodes, positioned in specific locations inside the building, which provide directions regarding the best available exit. Each decision node executes periodically a distributed algorithm that calculates the best direction towards a building exit, by using only local information coming from neighbouring sensor and decision nodes. The outcome of the calculation can be communicated to the evacuees via smart panel indicators or portable wireless devices they carry. We evaluated the performance of the system using a multi-agent building evacuation simulator we developed. Our simulation experiments include various evacuation scenarios which take place inside a three-storey building. We also compared our results against the case where the system is not used. The simulation results show that the overall evacuation outcome is improved by the use of the decision support system. More specifically, thanks to the decision support system the civilians are able to reach an exit faster, by avoiding the exposure to the hazard.

In future work we will propose modifications of the effective length formulation, by taking into account the unique characteristics of our problem. We will investigate how we can extend our algorithm in order to exclude paths that can become potentially hazardous in the near future. Moreover, we will study different algorithmic approaches to the problem of the safe evacuation of civilians and propose the use of different metrics for measuring the safety of a path towards a building exit.

8. ACKNOWLEDGEMENTS

This research was undertaken as part of the ALADDIN (Autonomous Learning Agents for Decentralised Data and Information Networks) project and is jointly funded by a BAE Systems and EPSRC (UK Engineering and Physical Research Council) strategic partnership (EP/C548051/1).

REFERENCES

[1] R. R. M. Gershon, "The world trade center evacuation study: Lessons for other high rise office buildings," in *NFPA World Safety Conference & Exposition*, June 2006.

[2] E. Gelenbe, E. Seref, and Z. Xu, "Simulation with learning agents," *Proceedings of the IEEE*, vol. 89, no. 2, pp. 148–157, Feb 2001.

[3] S. G. S. Batalin M. and H. M., "Mobile robot navigation using a sensor network," in *IEEE International Conference on Robotics and Automation*, April 2004, pp. 636–642.

[4] P. Corke, R. Peterson, and D. Rus, "Networked robots: Flying robot navigation using a sensor net," in *11th International Symposium of Robotics Research (ISRR 2003)*. Springer-Verlag, October 2003, pp. 234–243.

[5] Q. Li, M. D. Rosa, and D. Rus, "Distributed algorithms for guiding navigation across a sensor network," in *MobiCom '03: Proceedings of the 9th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2003, pp. 313–325.

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

[7] D. Bertsekas and R. Gallager, *Data networks*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1987.

[8] P. A. Humblet, "Another adaptive distributed shortest path algorithm," *IEEE Transactions on Communications*, vol. 39, pp. 995–1003, 1991.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Second Edition*. The MIT Press, September 2001.

[10] E. Gelenbe, R. Lent, and Z. Xu, "Measurement and performance of a cognitive packet network," *Journal of Computer Networks*, vol. 37, no. 6, pp. 691–701, Dec. 2001.

[11] A. Filippoupolitis, L. Hey, G. Loukas, E. Gelenbe, and S. Timotheou, "Emergency response simulation using wireless sensor networks," in *Ambi-Sys '08: Proceedings of the 1st international conference on Ambient media and systems*, 2008.

[12] A. Filippoupolitis, E. Gelenbe, D. Gianni, L. Hey, G. Loukas, and S. Timotheou, "Distributed agent-based building evacuation simulator," in *Summer Computer Simulation Conference (SCSC '08)*, 2008.

[13] F. L. Bellifemine, G. Caire, and D. Greenwood, *Developing Multi-Agent Systems with JADE (Wiley Series in Agent Technology)*. Wiley, April 2007.

[14] D. G. Elms, A. H. Buchanan, and J. W. Dusing, "Modeling fire spread in buildings," *Fire Technology*, vol. 20, no. 1, pp. 11–19, 1984.