Emergency Response Systems for Disaster Management in Buildings

Avgoustinos Filippoupolitis, Georgios Loukas, Stelios Timotheou
Nikolaos Dimakis and Erol Gelenbe

{afil, gl1, stelios.timotheou
nikolaos.dimakis07, e.gelenbe}@imperial.ac.uk
Intelligent Systems and Networks Group
Dept. of Electrical & Electronic Engineering
Imperial College London

Abstract. Emergency response operations can benefit from the use of information systems that reduce decision making time and facilitate coordination between the participating units. We propose the use of two such systems and evaluate them with a specialised software platform that we have developed for simulation of disasters in buildings. The first system provides movement decision support to evacuees by directing them through the shortest or less hazardous routes to the exit. It is composed of a network of decision nodes and sensor nodes, positioned at specific locations inside the building. The recommendations of the decision nodes are computed in a distributed manner and communicated to the evacuees or rescue personnel in their vicinity. The second system uses wireless-equipped robots that move inside a disaster area and establish a network for two-way communication between trapped civilians and rescuers. They are autonomous and their goal is to maximise the number of civilians connected to the network. We evaluate both proposed information systems in various emergency scenarios, using the specialised simulation software that we developed.

1 Introduction

Information systems designed for emergency response operations can provide invaluable help for better planning and coordination during an ongoing crisis. Here, we focus on emergencies that take place inside a building. The time-critical nature in such situations necessitates fast decision making and reliable communication with emergency personnel and rescuers. For example, a crucial aspect of disaster management inside a building is the safe and rapid evacuation of its occupants. The evacuees often are not aware of the optimal evacuation route or may just not follow it. This is particularly common when there is an ongoing hazard, such as smoke, fire or flooding. Thus, we start by proposing a distributed system that computes the best evacuation routes in real-time, while a hazard is spreading inside the building. Also, not all the occupants of a building will be able to move without assistance during a disaster. There may be incapacitated or trapped civilians waiting for rescuers to reach them. Until this happens, it would be particularly beneficial to have some form of communication between rescuers and civilians, but usually the existing communication infrastructure is inadequate. Thus, we also propose the use of wireless-equipped robots that can establish an ad hoc network with trapped civilians. This network may provide voice or video connection and vital sensor data that help the rescuers better assess the situation and plan their actions. We implement and evaluate both systems using a multi-agent simulation platform that we developed. The use of simulation gives us the opportunity to test our systems in numerous disaster scenarios, while being able to vary parameters such as the hazard location, the number of evacuees, the locations of the victims and the building structure.

The remaining of the paper is structured as follows. We first present an overview of the simulation platform that we used for the evaluation of our proposed information systems. Then, we continue with the movement decision support system for evacuation and with the robotic network system for communication with trapped civilians. We conclude with a summary of our contributions and potential directions for future work.

2 Evacuation Simulation Platform

Our simulation environment is based on the JADE platform, which is a software platform for developing applications in compliance with the FIPA specifications for interoperable intelligent multi-agent systems [1]. The goal is to simplify the development while ensuring standard compliance through a comprehensive set of system services and agents. It is accompanied by a number of preferred features that are suitable for developing simulation environments, such as agent mobility. Our simulation platform relies on discrete event simulation techniques. All entities register to the controlling simulator and define the time at which they are to be awaken. The simulator undertakes the re-organising of the entities and triggers each agent to execute at its corresponding time. Furthermore, it is able to operate in real-time, which facilitates the integration of external components, such as a real sensor network.

Our agent structure is organised in a multi-layered approach, so that each layer provides specialised functionality and contributes to the evolution of the simulation. There are dedicated layers for controlling the basic functionality of the agent, such as behaviour management and agent registering and deregistering, maintaining a link with the simulation agent and accordingly compile and extract simulation-specific communication messages, and a layer which allows interaction with other agents. The disaster area is modelled as a graph or a collection of graphs, which contain special nodes, such as entrances and staircases. For example, figure 2 shows two areas with five collection points and three graph bridges. Each of these areas is controlled by a dedicated simulator, and simulator is aware only of its area of interest and how it is connected with other areas via the graph bridges. Similarly, each simulated entity has an initial perspective of the overall areas with a rough estimate of the edge lengths. As these

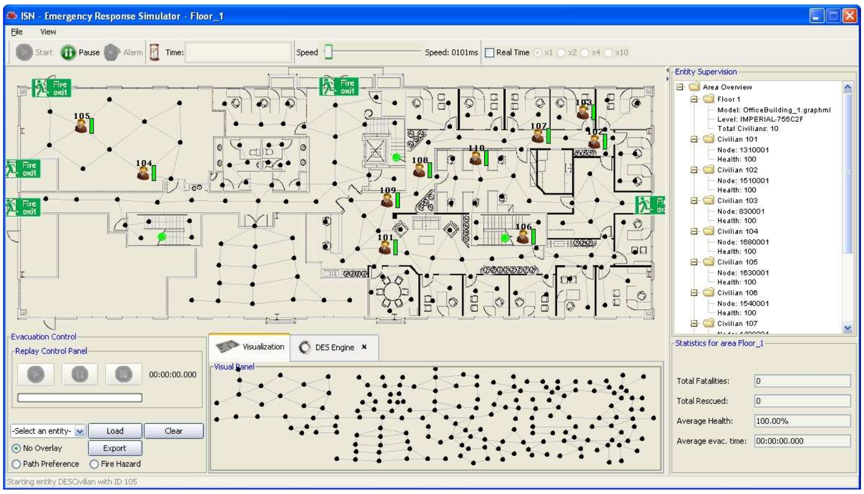


Fig. 1. Graphical interface of the simulation platform

agents move in the graph and interact both with the simulators but also with other agents, this perspective is updated with more accurate data. Using this graph-based approach, we are able to focus on different areas of the simulation.

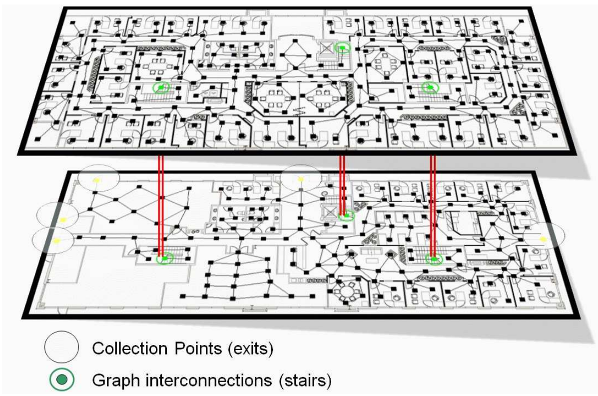


Fig. 2. Two areas with 5 collection points (exits) and 3 graph bridges (staircases)

The simulated actors are agents with individual characteristics. For instance, we can simulate a number of heterogeneous entities such as evacuating civilians, robots that explore the building, injured civilians, rescuers etc. The state of each entity is represented by its location, health level and an individual goal, which is typically the target location of its movement. As the simulation evolves, these parameters can be affected by the environment or other agents. For example, a fire spreading in a building will affect the health of some of the civilians and will block exits. Also, each entity has its own world perspective. This perspective is an initial estimate of the overall graph model which reflects the whole area under simulation. As the simulation evolves and the entities traverse the graph towards the exit or any other node, they update their internal perspective with the current surroundings.

3 On-Line Decision Support System for Building Evacuation

During crises in buildings, people are usually not aware of the optimal evacuation route and follow each other or often try to exit from where they entered. Moreover, if there is an ongoing hazard that is spreading in the building, then the best evacuation routes may change in the course of the evacuation procedure. We propose a system that computes the best evacuation routes in real-time and informs the evacuees accordingly. The system consists of a number of Decision Nodes (DN) installed in specific locations in the building. Their role is to provide directions to the evacuees regarding the best available exit. There is also a network of sensors that provides the DNs with real-time information regarding the conditions in the building. An underlying communication network links the DNs and the sensors. The recommendations of the DNs are computed in a distributed manner, at each DN, and are then communicated to the evacuees or emergency personnel. This can be achieved in the form of "smart panel" indicators installed on the DNs, or as information sent wirelessly from the DNs to handheld devices. We have implemented the proposed decision support system in the simulation software presented in Section 2. Figure 3 depicts a simulation scenario where the decision support system is in use. A fire is spreading inside the building floor while the occupants try to find the best available exit. The arrows correspond to directions from the smart panel indicators positioned inside the building. Each arrow points toward the direction a civialian should follow in order to reach the best available building exit.

There are various approaches regarding the problem of decision support for emergency situations. In [2], the authors propose a system based on wireless sensor nodes, that can navigate a robot or a human towards an exit by avoiding the hazardous areas. Their approach, however, assumes a static hazard and there is no evaluation for a simulation scenario that includes evacuees who use the proposed system. A similar system is proposed in [3], which uses a sensor network in order to calculate a path that leads to an exit and does not pass through the hazardous area. The authors demonstrate the ability of the algorithm to find the safest paths, but do not consider a spreading hazard nor evaluate their approach in a simulation scenario that involves evacuees. Our approach has been inspired by the work presented in [4] where vehicles, modelled as smart agents, are traversing a dangerous urban grid. The agents, who use information coming from the environment and from other agents, are able to adapt in order to travel rapidly and safely. Our system operates in a building environment, where

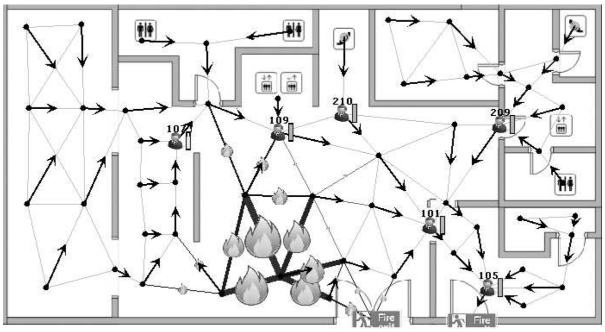


Fig. 3. The decision support system implemented in the evacuation simulation platform. The arrows correspond to directions from the "smart panel" indicators.

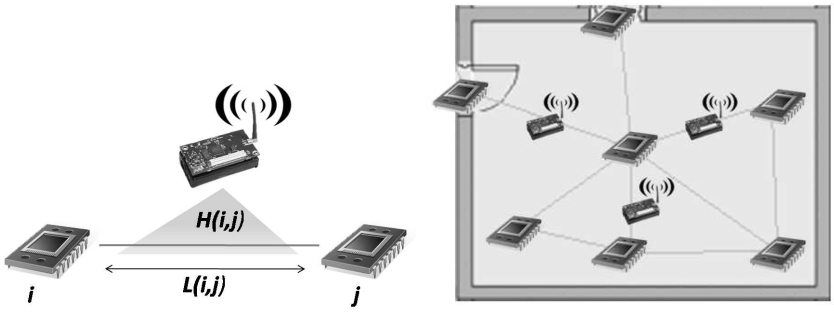
civilians are taking part in an evacuation in the presence of a spreading hazard. By following the directions provided by the decision support system, they can evacuate the building using the best available paths and avoiding the hazardous areas.

In our approach, we have assumed a known building layout. This is valid for the case of a decision support system that has been pre-installed in the building before the evacuation process is initiated. Another assumption is the existence of a number of DNs, installed in specific locations inside the building. These devices do not need to have high processing power or storage. Their job is to compute the direction that should be followed by each evacuee, towards the best available exit. The advice of a DN is communicated to people in its vicinity, by the use of a visual indicator (such as a smart panel) or by a wireless communication device (i.e. a PDA) which is carried by the evacuees or the emergency personnel. Finally, we assume the existence of a network of sensor nodes, that provides the DNs with real-time information about the conditions inside the building. This information can be related to the temperature of a location or the presence of smoke.

The known building layout is used in order to construct a graph G. The vertices of the graph correspond to locations where people can congregate (e.g. rooms, corridors, doorways or hallways). A link between two vertices of the graph represents a path that can be followed by the evacuees. The length $l(i,j)$ of a link between two vertices 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 with the observed hazard.

In order to obtain a metric that expresses how hazardous a link is, we introduce the idea of "effective length". We define the effective length $L(i,j)$ of a link as:

$$L(i,j) = l(i,j) \cdot H(i,j) \quad (1)$$



(a) (b)

Fig. 4. (a)A sensor that monitors hazard intensity on a link between two Decision Nodes, (b) Decision Nodes and Sensors Nodes Positioned in specific building locations

As we can see from Equation 1, the value of L 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, we have $L \equiv l$, since the effective length is equivalent to the physical length of the link. The higher the value of L on a link, the more hazardous it is for a civilian to move along it. Figure 4(a) illustrates the use of sensor nodes for determining the value of the effective length.

Figure 4(b) depicts an example topology where DNs and sensor nodes are positioned in specific locations inside a room of a building. Each DN is placed at each of the vertices of the graph G. In practice, however, there could be fewer DNs than vertices in G, with each DN being in charge of providing decisions for a set of contiguous locations of G.

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. The algorithm that we propose is inspired by the distributed shortest path problem [5–7] and from adaptive routing techniques such as Cognitive Packet Networks [8]. It is executed by each DN, in a distributed manner, and its output is the next DN that is on the best available path towards an exit.

A DN, 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 y from n to an exit e : $L(n, e, y)$
- The effective length of the shortest path x , from u to an exit e: $L(u, e, x)$
- The next suggested DN

The initial conditions for the algorithm are set as follows:

$$L(u, e, x) = \begin{cases} 0 & \text{, if } x \in E \\ \infty & \text{, otherwise} \end{cases} \qquad (2)$$

where E is the set of available building exits.

Since the decision support system is already installed in the building when the hazard starts spreading, we can consider that the initial condition for each DN,at that time, is the actual physical length $l(u, e, x)$ of the shortest path from a DN 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 DN 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 DN 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 DNs. Each DN 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 DN, the locations of the exits and the spreading of the hazard.

Algorithm 1 Distributed calculation for the effective length $L(u, e, x)$

Send to every neighbour n of u, the effective length of the path from u to the exit $e : L(u, e, x)$
for each sensor node monitoring a link incident to u **do**
  Request hazard intensity H from sensor node
  Calculate the effective lengths $L(u, n)$,where n is a neighbour of u
end for
Update the effective length $L(u, e, x)$ of the shortest path x to the exit:
$L(u, e, x) = \min \{L(u, n)+L(n, e, y)$: $\forall$ neighbours n of u, $x = ny\}$
Set the next suggested DN v:
$v = \mathrm{argmin}\ \{L(u, n) + L(n, e, y)$: $\forall$ neighbours n of u, $x = ny\}$

When the decision support system is in operation, each DN at u periodically executes Algorithm 1 and provides a suggestion to the evacuees that are in its vicinity. The suggestion is of the form **"go to v"**. Communication and computation is much faster than the movement of individuals. Since conditions will change rapidly (e.g. the spread of fire), the DNs will periodically execute the algorithm, update the distance information and communicate the most recent valid advice to neighbouring evacuees or other active entities such as firemen or rescue personnel.

We have evaluated the proposed decision support system using the evacuation simulation platform. In the case where the decision support system is used, each civilian decides its next destination based on the recommendation of the respective DN. The movement of the evacuees in the absence of the system is modelled using an optimistic approach: each evacuee is assumed to have a full knowledge of the building's structure before the hazard starts spreading, and he becomes aware of a hazardous area when it reaches a location close to it. In the evacuation scenario we consider the a three-storey building depicted in Figure 5.

The initial location of the fire is in the second floor of the building, while the exit is located on the ground floor. We have conducted two sets of simulation runs, each one with a different level of building occupancy. In the first set we have assumed a total of thirty civilians inside the building (ten civilians per floor), while in the second set we considered twenty civilians per floor. We tested out system for different cases of the algorithm execution frequency. We conducted two hundred simulation runs for each case, with different initial civilian locations and different fire spreading rates in each case. Figure 6 illustrates the percentage

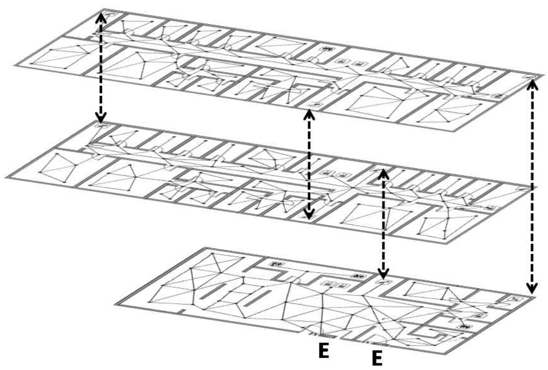


Fig. 5. The building used in the evacuation scenarios

of evacuees that have excited the building , versus the evacuation time. We can note that the outcome of all three 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: The decision support system achieves a faster evacuation and a higher percentage of safely evacuated civilians. We observe that the higher occupancy level of the building affects the total time of the evacuation, as it takes longer for the civilians to exit the building due to the increased congestion. However, as it can be verified by Figure 6(b), the use of the decision support system still results in a faster and safer evacuation. The average remaining health of the evacuees is shown in Figure 7, for the two different building occupancy levels. It is now more clear that the presence of the decision support system directs the evacuees away from hazardous areas and towards the best available exit. We notice that when the occupancy level of the building increases, the absence of the decision support system results in an even lower average remaining health. 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 spreads in more areas and the probability of being exposed to it increases. Figure 8 shows the percentage of fatally injured evacuees. We can again verify that the use of the decision support system results in minimal casualties during the emergency situation and that in every case it provides better results compared to the case where the system is not present. We must note that the algorithm execution frequency affects the performance of the system, with higher values giving better results. This is due the fact that the propagation of the changes in the environment (e.g. the change on the measurement of a sensor node) depend on the execution frequency. A high value for the algorithm execution frequency results in a more adaptive system which is able to give better suggestions to the evacuees.

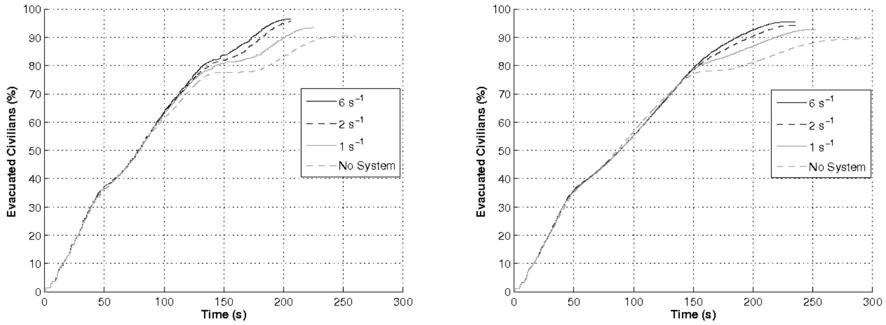
4 Robotic Networks for communication with trapped civilians

Mobile robots are routinely used in disaster management operations to reach areas that are inaccessible to humans. Usually, they are designed to search for victims, inspect the structural integrity of buildings, or detect hazardous materials, but with recent advances in small-size robotics and wireless communications, emergency response robots can also be used to form ad hoc networks. The following typical large-scale emergency situation indicates the usefulness of such a robotic network:

An earthquake has demolished a large building block in a city; the rescuers have arrived and need to assess the situation. Traditionally, the best case scenario is that the civilians use whistles or some more sophisticated radio-transmitting personal emergency device that facilitates their detection. From detection to rescue however, a long period may pass during which establishing and maintaining communication between the rescuers and trapped civilians is vital. During this period, the rescuers' job would be immensely assisted if instead of a simple notification device the civilians carried a device that would provide wireless connection with the rescuers, in the form of VoIP, live video streaming or even environmental and biomedical sensor data. In this way, the rescuers would be in position to better assess the health condition of the victims and the state of their local environment long before locating them. Given that the existing communication infrastructure may be partially or completely destroyed, a promising approach would be to employ mobile robots to act as wireless routers and form a network with the wireless devices of the trapped civilians (Fig. 9).

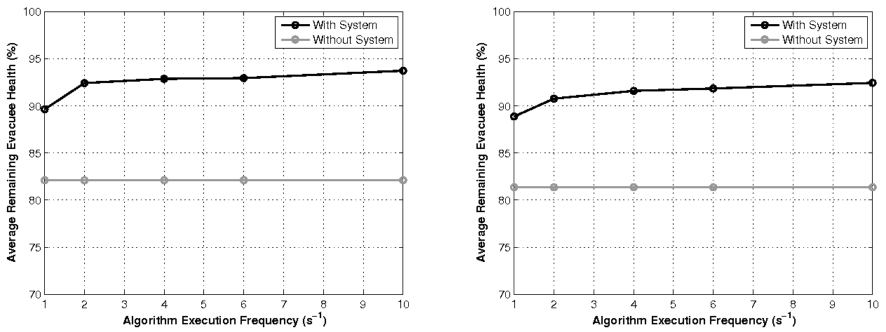
For this emergency communication paradigm, the fact that we have a limited amount of robots means that they need to be deployed efficiently to optimise different key objectives, such as time for the formation of the network or energy limitations. Yet, the most important objective is to maximise the number of civilians within range of the robotic network while maintaining multi-hop connectivity between the robots; this is the problem that we deal with here.

In this application we are investigating the use of robots equipped with wireless devices that move inside a disaster area to connect injured civilians with a wireless sink node. The latter represents the group of rescuers or the centre of operations. We assume that the civilians also carry a short-range wireless device for communication. This can be a dedicated personal emergency device, a bluetooth-equipped mobile phone, or some other wireless device. The goal of



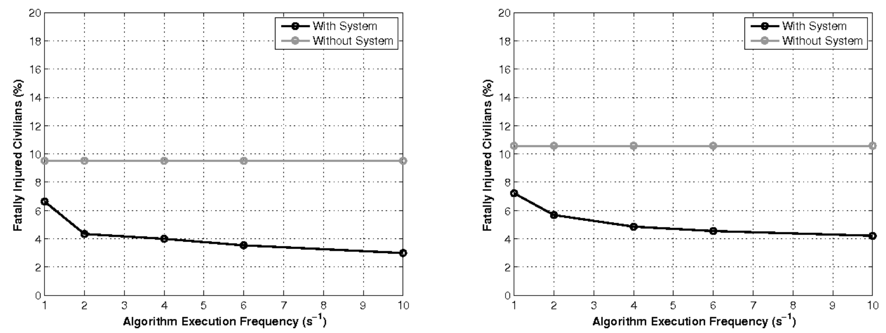
(a) (b)

Fig. 6. Percentage of evacuees that have exited the building vs. evacuation time, for different building occupancy levels: 10 civilians per floor (a) and 20 civilians per floor (b), and different algorithm execution frequencies



(a) (b)

Fig. 7. Mean remaining evacuee health vs. algorithm execution frequency, for different building occupancy levels: 10 civilians per floor (a) and 20 civilians per floor (b)



(a) (b)

Fig. 8. Mean percentage of fatally injured evacuees vs. algorithm execution frequency, for different building occupancy levels: 10 civilians per floor (a) and 20 civilians per floor (b)

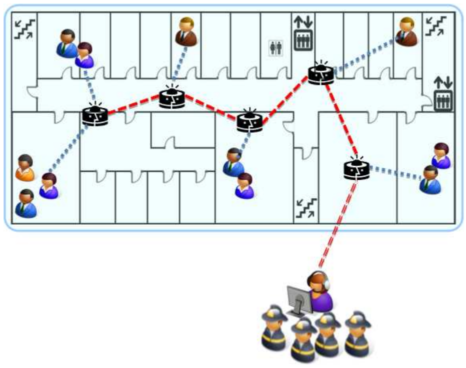


Fig. 9. Motivating scenario: A group of robots establish communication with trapped civilians

the robots is to maximise the number of civilians simultaneously connected to the network. A civilian is considered connected not only when he/she is directly in range with the sink, but also when in range with a robot that maintains multi-hop connectivity to the sink. We assume that two wireless entities, such as civilians or robots, are in communication range when the range of each entity is greater than their euclidian distance. We also assume that the robots have a priori knowledge of the disaster area.

When the robots are centrally controlled, we can formulate the problem with known locations as a NP-hard mixed-integer program which can be solved to optimality [9]. Although a centralised formulation provides an optimal solution, it may not be desirable in practice due to the time limitations of the emergency situation. For this reason we have developed a distributed algorithm in which the robots collectively try to optimise their performance using a set of predetermined rules and communication with each other. To test the algorithm, we have used the simulation platform presented in Section 2. The main challenge of the distributed approach, is that not only do the robots have to be efficiently deployed to connect as many civilians as possible, but also they must discover the civilians and cooperate to maintain connectivity of the formed wireless ad-hoc network. This can be significantly simplified if we consider that the civilians are naturally clustered in groups, either because they were together when the disaster occurred or grouped with others in their effort to survive. We exploit this by clustering the locations of civilians so that their maximum radius is smaller than $R_{rob} + R_{civ}$, because then by locating a robot at the centre of this cluster, the connectivity constraint is always satisfied (Fig. 10). The robot that settles on the cluster centre acts as a cluster leader and is responsible to issue an exploration announcement to all available robots in its range, which in turn connect the civilians of this cluster. Between clusters, chains of robots are formed to ensure connectivity.

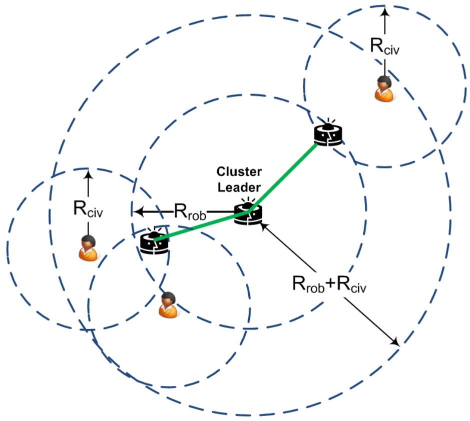


Fig. 10. Connectivity is guaranteed within a cluster if its radius is smaller than $R_{rob} + R_{civ}$

A high-level representation of the algorithm can be seen in Fig. 11. Within each cluster, the robots are allocated according to the number of civilians they will connect.

Essentially, our heuristic approach is composed of two stages:

- Move to most attractive cluster of civilians forming a chain of robots to maintain connectivity between clusters
- Connect the civilians of this cluster and move to the next one

Each of the robots greedily selects the cluster to which it is attracted the most. Several attractiveness metrics can be used such as the number of civilians in each cluster, the distance between the robot and each cluster, or a combination of the two. For the sake of simplicity we use the ratio of these two metrics so as to maximise the number of connected civilians and minimise the number of robots that settle to maintain connectivity between clusters.

In order to avoid having multiple robots at the same location, each one reserves the location where it intends to settle to act as a cluster leader, to connect civilians, or to maintain multi-hop connectivity between cluster leaders. A robot does not reserve a location from where it would lose connectivity, and this ensures that the final robotic network will be connected.

Connecting the civilians to the robotic network is done in a greedy fashion. Taking into consideration the already reserved locations of robots and the civilians that these robots connect, a robot selects a location in the cluster where it connects the maximum number of the remaining civilians.

We evaluated this algorithm as the movement decision model of robot agents in the evacuation simulation software. The results of the distributed algorithm are comparable to the centralised approach in terms of the number of civilians connected to the network (Fig. 12).

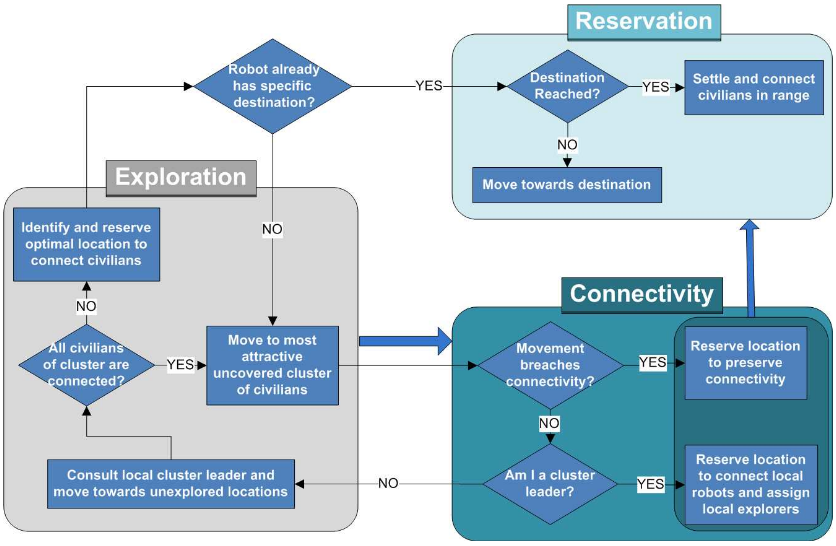


Fig. 11. General overview of the distributed algorithm

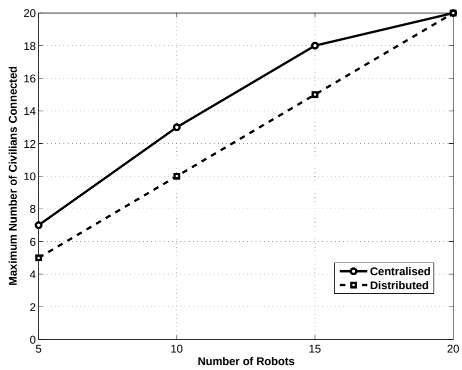


Fig. 12. Comparison between the distributed and centralised approach in terms of number of connected civilians against the number of robots

5 Summary and Future Work

We have proposed the use of two systems that can facilitate emergency response operations during an ongoing crisis in a building. The first system provides directions to evacuees regarding the best exit route. It consists of decision nodes that are positioned at specific locations inside the building and sensor nodes that provide information related to the hazard. Each decision node uses only local information to compute the best direction towards the exit in a distributed manner and communicates its result to the evacuees via smart panel indicators or wireless devices. The simulation results illustrate that the decision support system improves the outcome of the evacuation procedure by directing the evacuees along safer paths. In future work, we will take into account additional parameters, such as prediction of the congestion and of the dynamic propagation of a hazard. We will also investigate the impact of network characteristics, such as delay and packet loss, on the performance of the system.

We have also proposed the use of autonomous robots that move inside a disaster area and establish a wireless network for two-way communication between trapped civilians and an operation centre. We presented a distributed algorithm that is run on each robot so that they collectively maximise the number of civilians connected to the network by clustering possible locations of civilians. This work opens the way for a number of new research challenges. For example, the employment of clusters for civilian exploration and connectivity leads to interesting optimisation problems, such as the optimal exploration choices within a cluster to minimise the exploration time or the energy expenditure. Finally, robust approaches that take into account any robot or communication failures should be developed to ensure the uninterrupted connectivity of the robotic network.

In the future, we intend to address different aspects of crises in a building, such as the allocation of rescuers to injured civilians. We will also extend the evacuation simulation platform to evaluate a wider range of such information systems.

References

1. "http://www.fipa.org," Foundation for Intelligent Physical Agents (FIPA).
2. 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.
3. 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.
4. E. Gelenbe, E. Seref, and Z. Xu, "Simulation with learning agents," *Proceedings of the IEEE*, vol. 89, no. 2, pp. 148–157, Feb 2001.
5. D. Bertsekas and R. Gallager, *Data networks.* Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1987.
6. P. A. Humblet, "Another adaptive distributed shortest path algorithm," *IEEE Transactions on Communications*, vol. 39, pp. 995–1003, 1991.

7. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms, Second Edition.* The MIT Press, September 2001.
8. 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.
9. G. Loukas, S. Timotheou, and E. Gelenbe, “Robotic wireless network connection of civilians for emergency response operations,” in *23rd of the International Symposium on Computer and Information Sciences (ISCIS 2008), Istanbul, Turkey*, 2008.