The Computer Journal Advance Access published June 15, 2012

doi:10.1093/comjnl/bxs063

Spatial Computers for Emergency Support

AVGOUSTINOS FILIPPOUPOLITIS*, GOKCE GORBIL AND EROL GELENBE

Department of Electrical and Electronic Engineering, Intelligent Systems and Networks Group, Imperial College London, London, UK
**Corresponding author: afil@imperial.ac.uk*

We present two spatially distributed computing systems that operate in a building and provide intelligent navigation services to people for evacuation purposes. These systems adapt to changing conditions by monitoring the building and using local communication and computation for determining the best evacuation paths. The first system, called distributed evacuation system (DES), comprises a network of decision nodes (DNs) positioned at specific locations inside the building. DNs provide people with directions regarding the best available exit. The second system, called opportunistic emergency support system (OESS), consists of mobile communication nodes (CNs) carried by people. CNs form an opportunistic network in order to exchange information regarding the hazard and to direct the evacuees towards the safest exit. Both DES and OESS employ sensor nodes deployed at fixed locations for monitoring the hazard. We evaluate the spatial systems using simulation experiments with a purpose-built emergency simulator called DBES. We show how parameters such as the frequency of information exchange and communication range affect the system performance and evacuation outcome.

Keywords: spatial computers; emergency navigation; building evacuation; opportunistic communications; cyber-physical systems

Received 15 December 2011; revised 28 March 2012
Handling editor: Jacob Beal

1. INTRODUCTION

Emergencies can be highly dynamic situations that require quick and correct decision-making. The use of autonomous emergency support systems, such as evacuation guidance, can improve the outcome of an emergency situation by providing improved situational awareness and added services to civilians and responders in the affected area. In this paper, we specifically consider the problem of emergency evacuation of confined spaces.

Evacuation of a building during an emergency situation, such as a fire, is a complex and challenging task. The occupants have to quickly decide which path to follow in order to exit the building safely. As conditions in the building change due to the spreading of the hazard, it becomes difficult for an evacuee to find the best evacuation path. One of the main problems that evacuees face during an emergency is the lack of knowledge regarding the conditions in other parts of the building [1]. Most of the times there is ambiguous information with respect to which evacuation paths are safe and which locations are affected by the hazard. This can lead to a delay in commencing the evacuation of the building and can also result in choosing an inappropriate evacuation route.

A case that demonstrates how the lack of information can put lives in danger is presented in [2]: a fire starts in an apartment on the seventh floor of a residential building. A family of four residing on the 21st floor choose an unsafe path when they start to evacuate due to the lack of information on the hazard. This wrong choice causes them to backtrack when they encounter heavy smoke, and their efforts to find a new path are fruitless since the hazard had significantly spread by that time, blocking many escape paths that were initially available. Two members of the family eventually perish due to exposure to smoke during their evacuation attempt. This is one of many examples that illustrate the importance of accurate information and assurance given by an evacuation support system.

In this paper, we present two spatially distributed computing systems that combine monitoring of the environment and local communications to provide adaptive directions for evacuation of confined spaces, such as multi-floor buildings. As described in Section 2, the distributed evacuation system (DES) is based on static nodes while the opportunistic emergency support system (OESS) mainly consists of mobile nodes. These systems therefore represent different spatial structures for emergency support systems.

The rest of the paper is structured as follows: we present the design for our proposed spatial computing systems in Section 2. Evaluation of the systems with respect to different parameters is presented in Section 3, while Section 4 discusses related work in the area of emergency navigation and evacuation. We conclude with a summary of our contributions and possible extensions in Section 5.

2. DESCRIPTION OF THE SPATIAL COMPUTING SYSTEMS

We propose two spatial computing systems that provide adaptive evacuation directions to people in confined spaces, such as buildings, during an emergency situation. These systems comprises small, self-contained and self-powered computing devices spatially distributed inside the area of interest. Consequently, their design is closely related to the spatial characteristics of the operational environment. Therefore, we first state our assumptions regarding the physical environment and then proceed with the description of the systems.

2.1. Design assumptions

We represent the building as a graph $G(V, E)$, where vertices V are locations where civilians can congregate, such as rooms, corridors and doorways, and edges E are physical paths that civilians will travel along when moving inside the building. An example graph for a building floor is given in Fig. 1. The length $l(i, j)$ of an edge is the physical distance between vertices $i, j \in V$ while $H(i, j)$ represents the hazard intensity along this edge. We define the 'effective' length $L(i, j)$ of an edge as $L(i, j) = l(i, j) \cdot H(i, j)$. This metric expresses how hazardous an edge is for a civilian who traverses it. When there is no hazard along the edge, $L \equiv l$ and the effective length is equivalent to the physical length of the edge. As the value of H increases, the corresponding edge becomes more hazardous to traverse. The range of H can be defined based on the measured qualities of the hazard, such as temperature, light intensity and CO_2 level. In our simulations, we use nine discrete levels to describe the hazard intensity:

$$H = \begin{cases} 1 & k = 0 \text{ (no fire on edge)}, \\ k \cdot 10^3 & k = \{1, 2, \ldots, 8\}. \end{cases}$$

We assume that the graph is known for a building. We also assume that there are sensor nodes (SNs) installed in the building, where each SN monitors its immediate environment as depicted in Fig. 2. A sensor can potentially monitor multiple edges in the building graph based on its sensing capabilities and location. In our simulations, we assume that each SN monitors a single edge. When requested, an SN sends its latest measurement for its edge (i.e. its $H(i, j)$ value).

Each SN has a unique device ID, a location tag that represents the area (i.e. edge) monitored by the sensor and a short-range wireless communication capability. SN measurements are relayed to decision nodes (DNs) when the DES is in use, and to communication nodes (CNs) when the OESS is in use. Since SNs do not need to perform complicated computational tasks, their design can be kept very simple, with little memory and low processing power for cheap production. SNs are self-powered for a variety of reasons, including but not limited to low installation cost, ease of installation and maintenance and resilience to power outages. Since SNs are designed as 'deploy and forget' devices, meaning that their maintenance will be infrequent, and their energy use should be kept low. This is achieved by operating them in a sleep-duty cycle in non-emergency conditions and using very low transmission power for communications.

The region monitored by an SN is generally in the form of a circle. In order to accurately map SNs to the area graph, care should be taken in the construction of the graph and the placement of SNs so that each SN is placed near the centre of the

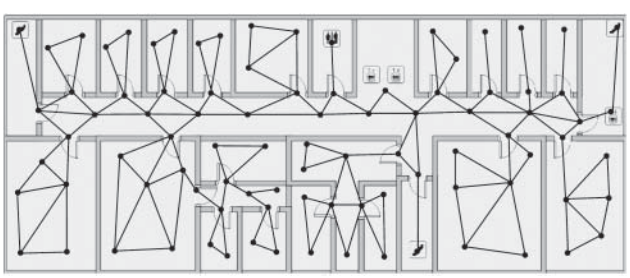

FIGURE 1. The floor plan of a building level and its graph representation. Vertices are locations where civilians can congregate and edges are physical path segments for civilian movement.

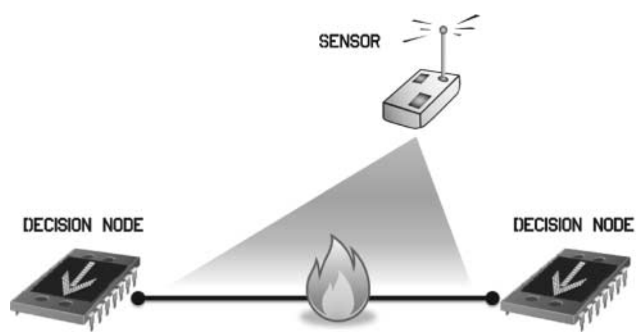

FIGURE 2. An SN monitors a graph edge for possible hazards. In the DES, DNs located at graph vertices receive measurements from their adjacent SNs and provide dynamic directions during evacuation. In the OESS (not shown), mobile CNs receive hazard measurements from SNs opportunistically as they move inside the building.

edge it monitors. In this way, each region monitored by an SN is approximated by a graph edge. When such a region cannot be accurately represented by a single edge, multiple adjacent edges could be used. Although an SN cannot sense across physical obstacles, such as walls, it can potentially communicate through obstacles. Since this could affect the DES operation, we elaborate on how it is addressed in Section 2.2.

Current technology allows the realization of SNs as used in our systems. A multitude of wireless sensor devices with different sensing and communication capabilities are available from many producers; an example is the TelosB wireless sensor [3], which is designed mostly for experimentation. Each TelosB device is equipped with a low-power, low-rate, short-range wireless communication module based on the IEEE 802.15.4 standard, and temperature, humidity and light sensors, which can be used to monitor the environment for simple hazards, such as fire and smoke. A more generic solution is provided by the MicaZ platform [4], where the basic communication and computation unit can be coupled with a variety of sensor modules as needed. Sensor modules more specific to fire monitoring are available in the form of smoke detectors [5] and CO/CO_2 sensors [6, 7] that can be linked with micro-controllers.

2.2. Distributed evacuation system

The DES consists of static DNs, which are installed at specific locations inside the building. In this discussion, we assume that a DN is positioned at each graph vertex. In practice, however, there could be fewer DNs, with each DN in charge of providing directions for a region consisting of multiple locations. Each DN is a battery-powered unit that has a short-range wireless communication capability, a processor and memory and a dynamic visual panel to display directions to civilians in the vicinity of a DN. Note that if a person is equipped with a wireless hand-held device (such as a smartphone), then DNs can also communicate their directions via a wireless link with such devices.

The operation of the DES is closely related to the system's spatial structure, since each DN computes the best direction towards the DNs located at building exits and communicates this (visually or via wireless) to the evacuees in its vicinity. DNs form a wireless network to exchange information in a distributed manner, so they can provide up-to-date dynamic evacuation directions to civilians as the hazard spreads. Hazard information is provided to DNs by their adjacent SNs (Fig. 2), and this information is further propagated among DNs based on the distributed decision algorithm presented below. Figure 3 depicts the DES as simulated by DBES (see Section 3.1), with each arrow representing the direction towards the best exit as given by a DN.

In the DES, a DN is interested only in measurements from SNs adjacent to it. Since wireless signals can propagate through walls, it is possible that a DN receives measurements from a non-adjacent SN (e.g. an SN behind a wall in another room). A DN can detect a non-adjacent SN based on its location (vertex) and the location of the area monitored by the SN (edge). Such measurements are not used and are discarded by the DN.

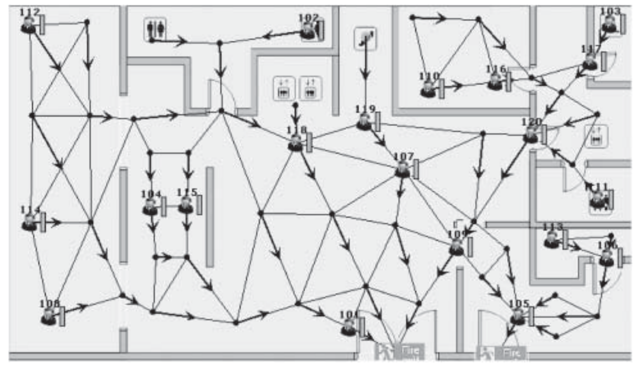

FIGURE 3. The DES as simulated by DBES. Arrows represent the directions suggested by the DNs. There is one SN per edge (not shown).

The definition of the effective length $L(i, j)$ only takes into account the hazard present along an edge (i, j) as monitored by the associated SN [8]. We can, however, take advantage of the spatial characteristics of hazard propagation inside a confined space to improve the performance of our system. Our approach is inspired by the fact that a hazard, such as fire or smoke, at a location will eventually affect neighbouring locations [9, 10]. When DNs use hazard information obtained only from their neighbouring SNs (i.e. their adjacent links), hazard values from nearby SNs (non-adjacent links) are not taken into account during path calculation. This results in the calculation of paths that are not yet hazardous, but will be in the near future due to the nearby hazard. When the hazard spreads to nearby locations and therefore to the paths that pass through them, DNs will calculate a different escape path for the evacuees. Waiting for the hazard to spread to calculate a new evacuation path can cause an increase in evacuation time since the evacuees may have to head to a new destination. Moreover, their health will be affected as they can potentially be exposed to the hazard during the process. By incorporating spatial hazard information in the effective edge length, the algorithm can proactively exclude paths that involve travelling near hazardous areas. To achieve this, we let each SN communicates with its neighbours and incorporate their readings into a 'spatial' hazard value H_{sp} that is reported. The number of neighbours with which a sensor can communicate is affected by the SN communication range R. Figure 4 shows an SN that communicates with its neighbouring SNs as determined by R. Note that when spatial hazard information is used, a DN indirectly receives measurements from non-adjacent edges. In our evaluation of DES, we will observe that the sensor communication range R has an important effect on the evacuation performance of DES.

To represent the modified effective edge length that incorporates spatial hazard information, we define $L_{sp}(i, j)$ for

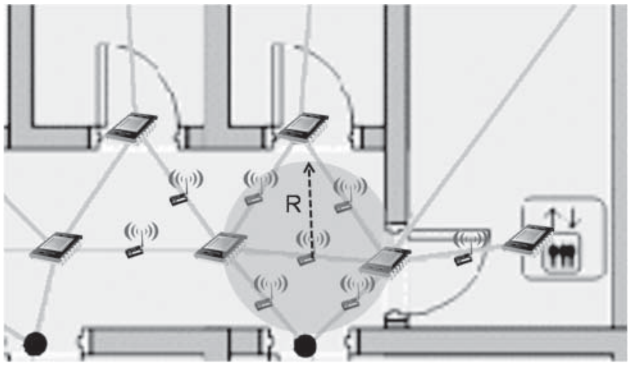

FIGURE 4. Incorporating spatial information into sensor measurements. This figure shows an SN and the sensors in its neighbourhood used to calculate the spatial hazard values. The neighbours are determined by the SN communication range R.

an edge (i, j). Let m be a sensor measuring the hazard level $H_m = H(i, j)$ on link (i, j). A sensor n measuring the hazard level H_n on a link (i', j') will then belong to the neighbours set $N(m)$ of m when $d(m, n) \leq R$, where $d(m, n)$ is the Euclidean distance between n and m and R is the communication range of the SNs. The effective length $L_{sp}(i, j)$ that includes the spatial hazard information is given by $L_{sp}(i, j) = l(i, j) \cdot H_{sp}(i, j)$, where

$$H_{sp}(i, j) = H(i, j) + \frac{1}{|N(m)|} \sum_{k \in N(m)} H_k.$$

Each DN, positioned at vertex u, stores the following information:

(i) its location in the building, u;
(ii) the effective edge lengths to neighbours: $L_{sp}(u, n)$, $\forall n \in V \mid (u, n) \in E$;
(iii) the effective lengths of the paths to an exit for all neighbours: $L_{sp}(n, e)$, $\forall n \in V \mid (u, n) \in E$ and e is a building exit;
(iv) the effective length of the shortest path (SP) from u to an exit e: $L_{sp}(u, e)$;
(v) the next suggested DN d (i.e. the next hop along the SP from u to an exit).

It is not necessary for a DN to keep information regarding the effective length of the paths to all available exits. As the algorithm is executed, this information is propagated from the exits to the DNs and each DN will eventually select the best exit. The distributed decision algorithm, given in Algorithm 1, is executed periodically by each DN. The algorithm is based on principles developed in [11, 12], and inspired by the distributed SP algorithm [13] and adaptive routing techniques such as the cognitive packet network [14]. Its output is the next hop (i.e. DN) towards the nearest building exit. As edge costs are a combination of physical distance and hazard intensity, the paths calculated by the DES minimize the travel distance to the exits while avoiding dangerous areas in the building.

Algorithm 1 Distributed calculation of the next hop to the best exit by DN u.

Send to every neighbour DN n of u, the effective length of the path from u to the exit $e : L_{sp}(u, e)$
for each SN monitoring a link incident to u **do**
  Request hazard intensity H_{sp} from SN
  Calculate the effective length $L_{sp}(u, n)$, where n is a neighbour of u
end for
Update the effective length $L_{sp}(u, e)$ of the shortest path to the exit:
$L_{sp}(u, e) = \min \{L_{sp}(u, n)+L_{sp}(n, e)\colon \forall \text{ neighbours } n \text{ of } u\}$
Set the next suggested decision node v:
$v = \operatorname{argmin} \{L_{sp}(u, n) + L_{sp}(n, e)\colon \forall \text{ neighbours } n \text{ of } u\}$

The initial values for the effective length at each DN depend on its location inside the building. Exit DNs set their $L(u, e)$ values to 0, while all other DNs set it to ∞. The output of the algorithm is the next hop (DN) along the path from the current DN to the best (safest and nearest) exit. The next hop is communicated to people in the vicinity of the DN via visual signals or wireless communications.

Note that DES is a spatial computing system with static sensing, processing and communication components. The spatial characteristics of the building affect where these components are located. System parameters, such as the SN range R and the execution frequency of Algorithm 1 by each DN, affect the performance of the system; we discuss the effects of these parameters in Section 3.3. We should also note that the DES is a distributed self-stabilizing system [15]. Since the DNs execute a distributed algorithm, convergence is achieved after a finite number of execution cycles. However, the values reported by the SNs can change due to the spreading of the hazard, before the algorithm has converged. In such a case, the DES is able to continue the distributed computation of the safest path without the need of reinitializing the algorithm. This characteristic allows us to avoid the use of a mechanism for reinitializing the algorithm in all the DNs, which benefits the overall system performance and avoids unnecessary communication overheads.

2.3. Opportunistic emergency support system

The OESS consists of mobile CNs carried by civilians. Each CN is a simple pocket device with a short-range wireless communication capability, a processor and local storage. CNs form an opportunistic network (oppnet) [16] that exploits node mobility to communicate over multiple hops. Such opportunistic communications (oppcomms) are characterized by the 'store–carry–forward' paradigm [17] where messages received

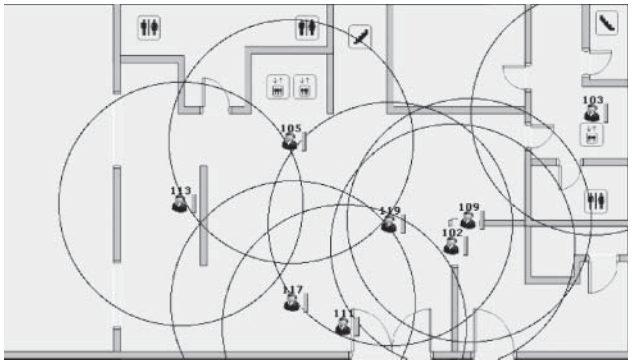

FIGURE 5. The OESS as simulated by DBES. Circles represent the maximum CN communication range.

by a CN is stored in local memory and carried with the CN as a result of human mobility. Messages stored on behalf of others are then forwarded to other CNs as they come into contact. Thus, a message is delivered to its destination via successive opportunistic contacts. Because the oppnet can be disconnected for long periods of time, CNs may need to carry messages for long durations and delivery of messages is not guaranteed.

Oppcomms are used to disseminate hazard information among CNs in the form of emergency messages (EMs). Hazard information is generated by SNs deployed in the building as described in Section 2.1. Each significant[1] hazard measurement is stored in a new measurement message (MM) created by the SN monitoring the affected area (e.g. edge). An MM contains the source ID (SN ID), location information (edge ID or (i, j)), the hazard intensity $H(i, j)$ and measurement timestamp. The latest MM created by an SN is forwarded to any CN that comes in contact with the SN. When an MM is received by a CN, it is used to update the local view of the CN as discussed below. The MM is also translated into an EM that contains the source ID (CN ID) and information from the MM (intensity, edge (i, j), timestamp). Multiple MMs are combined into a single EM when possible. In contrast to MMs, which are sent from SNs to CNs via single-hop communications, EMs are sent from CNs to CNs over multiple hops using oppcomms. Each EM is destined for all CNs. Figure 5 shows the OESS as simulated by DBES (see Section 3.1); circles represent the maximum communication range of CNs carried by the civilians.

The first MM or EM received by a CN acts as an alarm, indicating that there is a hazard and the user of the CN should evacuate the building. Each CN stores the building graph in local storage and received MMs and EMs are used by a CN to update edge costs on its local graph. An update triggers the calculation of SPs from the current CN location to all building exits, and the path with the lowest cost is used as an evacuation path. Any SP algorithm can potentially be used; CNs employ Dijkstra's SP algorithm [18]. Since effective edge lengths ($L(i, j)$ values) are used in SP calculation, the 'shortest' path minimizes exposure to the hazard while also minimizing travel distance.

The current evacuation path calculated by a CN is used to provide step-by-step directions to its user. In order to do this, the CN needs to know its location in the building. Indoor localization is achieved using the fixed SNs in OESS: each SN contains a location tag; we use the edge ID (i, j) monitored by the SN in this implementation as the SN location tag. Once notified of the emergency, each CN periodically sends a *beacon* using local broadcast. SNs that receive this beacon reply with a localization message (LM) that contains the source ID, location tag and timestamp. Very accurate localization is not required since the location of CNs are approximated by the graph vertices. The short communication range of CNs and SNs also decreases the localization error. The location of a CN is updated as it moves in the building via LMs, and at each location update the CN updates the directions given to its user based on its current location and evacuation path.

CNs use epidemic routing [19] for the dissemination of EMs, coupled with *timestamp-priority queues*, where EMs with the earliest creation timestamps are dropped from the queue when it is full. Although epidemic routing is an early oppnet routing protocol, our evaluations [16] have shown that it is very suitable for OESS due to its flooding-based approach. Epidemic routing is known to have high message delivery ratios and low message latencies at the cost of high communication overhead [20]. However, the communication overhead is not applicable in OESS since each message is targeted for all CNs, and good communication performance is desirable for emergency communications.

The OESS is a spatial computing system similar to DES, with the significant difference that the main computing and communication components of the system are mobile. This mobility introduces another level of dynamism to the system as communications are now affected by human mobility, which is in turn affected by oppcomms since CNs use oppcomms to provide dynamic directions to civilians. We evaluate the effect of one environmental and one system parameter, population density and CN range, respectively, on the performance of OESS discussed in Section 3.4.

3. EVALUATION OF THE PROPOSED SPATIAL COMPUTING SYSTEMS

We have evaluated our proposed systems by conducting simulation experiments of two multi-storey buildings using the DBES. In this section, we first provide an overview of DBES and then present our simulation model and assumptions. We then present our simulation results and discuss the evacuation performance of DES and OESS.

[1]Measurements that do not indicate abnormal conditions are ignored by SNs and do not result in MMs.

FIGURE 6. Graphical user interface (GUI) of DBES in interactive mode. DBES can be used in the batch mode with a command-line interface, or in interactive mode as presented here.

3.1. Distributed building evacuation simulator

We have used the DBES [21], developed by our research group in Imperial, for evaluating the proposed emergency support systems. DBES is an agent-based discrete-event simulation platform designed to support evaluation of emergency systems and scenarios in urban and confined spaces. It is a distributed platform that enables the execution of different parts of a simulation on multiple networked machines. Distributing a simulation allows quicker simulation of complex or large scenarios. DBES is based on the Java Agent Development Framework (JADE) [22], which is an open-source software platform for the development and implementation of multi-agent systems. Figure 6 presents the user interface of DBES in the interactive mode. Via this interface, users can control the simulation and alter the simulation parameters on-the-fly. Other emergency-related work that has employed DBES includes [23], which investigates the effect of group leaders in evacuation, and [24–26] where autonomous robots are used to form an *ad hoc* wireless network for emergency communications between responders and civilians.

DBES also supports augmented reality simulations where real-life hardware is integrated with software-based simulation; [23] presents an emulation where a wireless sensor network (WSN) of TelosB motes, discussed in Section 2.1, has been integrated with DBES. In that work, the WSN is used to monitor an artificial hazard, which is emulated using small lighting devices (LEDs). Light sensors on the motes are used to measure the hazard intensity, which is disseminated over the WSN to a gateway, and through the gateway into the simulation. In this paper, we model both the hazard and the SNs completely using software.

3.2. Assumptions and simulation model

As discussed in Section 2.1, we assume that the simulated area is represented as a graph. A person can occupy a single graph vertex at any time and can move to an adjacent vertex along an edge. Multiple people can occupy the same vertex at the same time. Each vertex has a first-in first-out queue that keeps the people currently located at that vertex. Physical congestion is modelled using a queuing approach, where each person waits until it is served by the vertex. The time it takes a person k to move from one vertex i to another j has two components: (i) the travel time, which is calculated as $l(i, j)/\upsilon(k)$, where $\upsilon(k)$ is the movement speed for k, and (ii) queuing time, which is based on the queue length at the time k arrives at i and the service time $\tau(i)$ for vertex i. Let $q(i, t)$ denote the queue length at vertex i at time t. Then k will have a movement time $\Delta t = \tau(i) \cdot q(i, t) + (l(i, j)/\upsilon(k))$, for an arrival time of $t + \Delta t$ at j. Person k resides at vertex i until $t + \Delta t$; at time $t + \Delta t$, she is removed from the front of the queue at i and added to end of the queue at j.

Note that this model has two parameters that can be arbitrarily set: υ and τ. The arrival rate of people depend on their movement. We set $\tau(i) = 1$ s, $\forall i \in V$, and $\upsilon(k) = 1$ m/s within floors and $\upsilon(k) = 0.7$ m/s at staircases, $\forall k \in P$, where P is the set of people in the building. The service time τ represents the amount of time needed by a civilian to receive visual information from a DN or a CN and make a decision on where to move next.

The hazard we evaluate in our simulations is fire and its associated effects, such as smoke. The hazard probabilistically spreads in the area along graph edges following a Bernoulli trial model and affects the health of civilians on adjacent vertices. The fire model and its effects have been inspired by Elms *et al.* [9] and Hasofer and Odigie [10]. The graph used for modelling hazard propagation can be different from the graph modelling the area. In that case, the hazard graph has to be mapped to the area graph. Using a different hazard graph allows the modelling of hazard propagation via paths not usable for civilian movement, such as through walls and floors. In these simulations, we use the the same graph to model the hazard and the area. We believe that using the same graph does not introduce any significant bias for the considered scenarios. The area graph captures physical obstacles, which are important factors that affect fire propagation in buildings, and it can therefore model accurately how the fire spreads.

Other factors may affect fire propagation and its effect on infrastructure. For example, utility networks in the building, such as electricity and natural gas distribution, can allow fire to spread faster and through unexpected paths. The effect of fire on a section of such a network can propagate to other parts of the same network, or to other dependent networks, in the form of failures. We are in the early stages of research on integrating utility networks in urban emergencies and leave the discussion of these as future work.

In each simulation, people are initially situated at random locations inside the building using a uniform distribution on the set of graph vertices. Civilians follow a probabilistic mobility model intended to simulate the movement of people during working hours when they are not evacuating the area. When a civilian is notified of the emergency, she follows directions provided by the DNs (with DES) or her CN (with OESS) in order to evacuate. Otherwise, she follows a SP-based evacuation model as described below. Note that psychological aspects are not taken into account in our simulations. For example, we assume that people act rationally and correctly follow directions provided by the DES or OESS. We also assume that a person starts to evacuate immediately when notified of the emergency. Each person acts individually and group behaviour or other aspects, such as helping or notifying others, are not considered. In previous work [23], we investigated the effect of group behaviour where people follow leaders during evacuation.

In these simulations, we assume that all devices operate correctly and there are no failures in the system. We have investigated the effect of failures on an evacuation outcome in [27].

When there is no evacuation system in use, we assume that civilians follow a SP evacuation model. This model assumes that people are familiar with the whole building, which translates into knowledge of the global area graph. We also assume that they know their location in the building at all times, and can calculate and follow the SP that leads to the nearest exit. A central alarm is assumed in this scenario and all people are alerted of the fire as soon as it starts and they start to evacuate the building immediately. If a person encounters the hazard during evacuation, she updates her knowledge of the building (i.e. the edge cost(s)) and re-calculates her SP. While the assumptions for SP evacuation are unrealistic, as they require too much from the evacuees during an emergency and do not consider issues like uncertainty and confusion, we believe that SP evacuation provides a valuable benchmark for comparison.

Each civilian starts with a health level of 100 units and her health decreases as she is exposed to the hazard; exposure duration, distance to the hazard and hazard intensity influence the effect of hazard on civilian health. A person with health ≤ 0 is assumed to have perished. In our current model, the health of a person does not affect her movement speed or ability to follow directions, although the DBES is certainly able to accommodate such effects. Evacuee health is currently used as a metric of evacuation performance as discussed in the following sections.

3.3. Simulation results: DES

We have evaluated the DES in an emergency scenario taking place inside the three-storey building depicted in Fig. 7. The dimensions of the building are $50\,\text{m} \times 18\,\text{m}$, and its graph is composed of 673 vertices and 775 edges. There are three staircases inside the building (represented by the dashed lines) which provide access to different floors. A fire erupts on the ground floor of the building and the occupants begin the evacuation of the building as soon as the fire starts, using the four exits located on the ground floor. The population density is 20 people per floor (20 pf). Presented results are an average of 100 simulation runs. The numbers in the legends of Figs 8 and 9

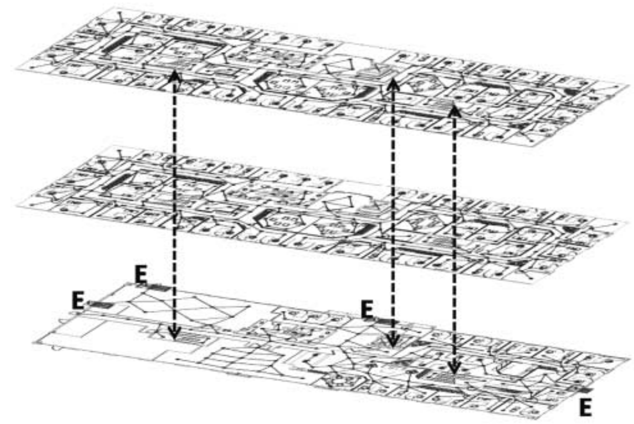

FIGURE 7. The three-storey building used for simulations of DES. The building is $50\,\text{m} \times 18\,\text{m}$ and its graph representation has 673 vertices and 775 edges. Dashed lines indicate staircases that connect floors. There are four exits on the ground floor.

represent the execution frequency of the distributed algorithm, measured in executions per second (Hz).

In the first set of simulations, we evaluate the effect of using spatial information in the hazard metric. The SN communication range is set to 2 m ($R = 2\,\text{m}$). Figure 8, shows that the distribution of evacuation times in the case with spatial hazard information is significantly different. This is due to the fact that the evacuees in the second case follow different paths while they try to exit the building. We can also verify that the total number of evacuees who exit the building is higher in the case where we use spatial hazard information. Moreover, we can notice that a higher algorithm execution frequency results in a lower evacuation time, since the DES is able to adapt faster to the dynamic conditions (hazard spreading) and provide the evacuees with accurate directions at an earlier stage. Finally, we should note that, when using DES, the evacuation procedure ends later compared with SP evacuation. This is due to the fact that when DES is used, a higher number of civilians manages to exit the building.

The percentage of fatally injured evacuees versus simulation time, given in Fig. 9, is complementary to the results presented in Fig. 8, and gives insight into how and when casualties occur during evacuation. Given that casualties are a result of exposure to the hazard, this is a metric on the effectiveness of DES in preventing exposure of civilians to the hazard during evacuation. We can first verify that the use of spatial information decreases the percentage of fatally injured evacuees. Furthermore, we must note that the time window ending at the instant of the last fatally injured evacuee is now larger. This effectively means that the evacuees 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.

DES decreases the exposure of civilians to the hazard differently based on the extent of the fire. Consider a person

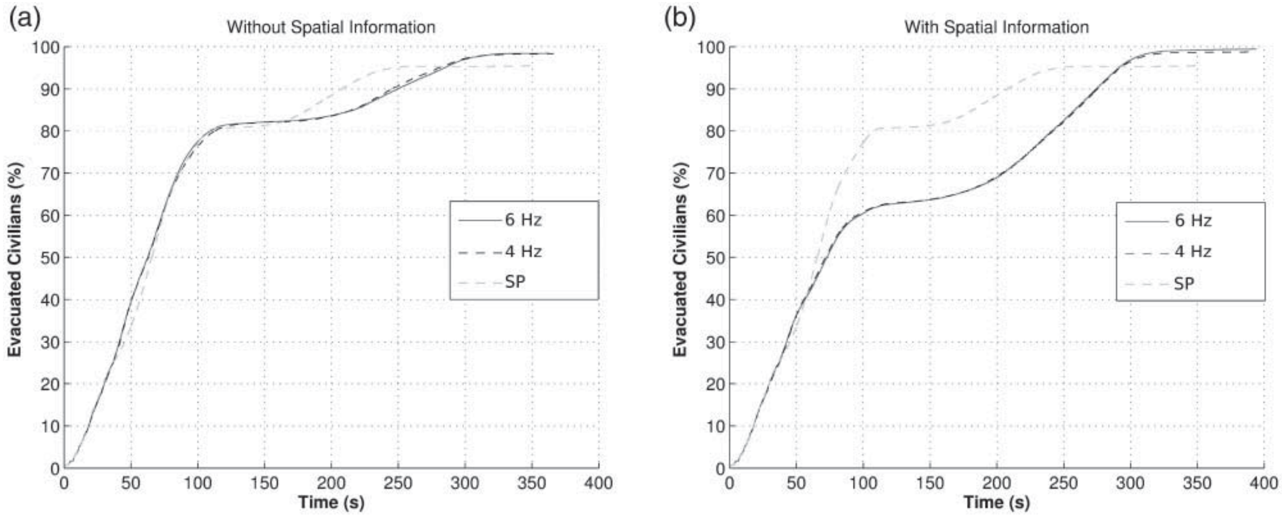

FIGURE 8. Percentage of safely evacuated civilians versus simulation time, with and without spatial information (DES).

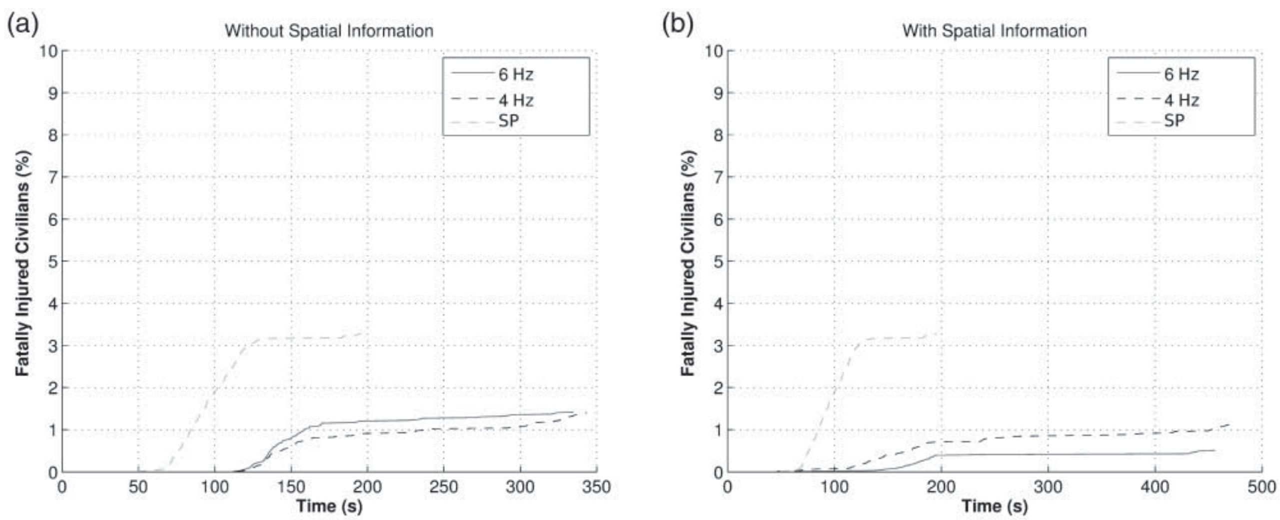

FIGURE 9. Percentage of fatally injured civilians versus simulation time, with and without spatial information (DES).

in the building, located at u. In the first scenario, there is at least one path unaffected by the hazard from u to an exit, and DES directs the person to the exit along the evacuation path. Since the person is navigated through areas largely unaffected by the hazard, exposure is decreased. In the second scenario, which is more likely to occur in the later stages of evacuation, there are no paths from u to an exit unaffected by the hazard. In this case, DES does not direct the civilian to an exit and risk exposure to the hazard, but instead directs her to a safe location in the building away from the fire. This behaviour is beneficial for cases where emergency responders are available since lifetime of the civilians trapped in the building is prolonged. It may, however, be a disadvantage when responders are unavailable or late to arrive. We will discuss the advantages and disadvantages of this approach in more detail when we compare OESS and DES in Section 3.4.

Note that this behaviour does not affect evacuation time since evacuation time is calculated based on people who successfully evacuate the building. However, it does affect the time of civilian casualties (as shown in Fig. 9) by prolonging their survival, even in the case where there is no safe path towards an exit and the occupants are effectively trapped inside the building. The real-time adaptation of DES to the spreading of the hazard is depicted in Fig. 9b. The shape of the curve for $t \in [100, 250]$ differs compared with that in Fig. 9a. This is a result of dynamic directions given to the civilians: as the fire blocks the current evacuation path, the DES calculates an alternative route and directs the evacuees towards an exit via the new path.

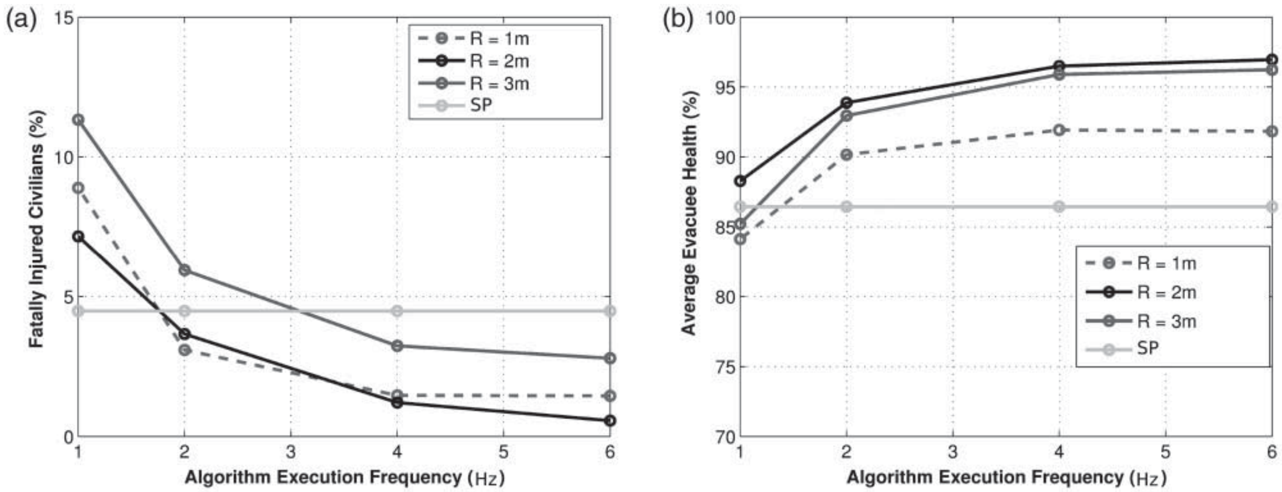

FIGURE 10. Average percentage of civilian casualties and average evacuee health versus algorithm execution frequency, for different SN ranges (DES). (**a**) Average civilian casualties. (**b**) Average evacuee health.

Simulation results regarding the performance of DES with different SN ranges and algorithm execution frequencies are shown in Fig. 10. We used three different ranges, $R = \{1, 2, 3\}$ m. The results obtained for the smallest range (1 m) are worse than the case where $R = 2$ m but better than the case $R = 3$ m. This indicates that there is a threshold value for R, until which increasing R increases the system performance due to the inclusion of larger spatial areas in the effective link cost L_{sp}. Beyond this threshold, the system performance deteriorates with increasing R. The reason for this behaviour is that DES gets more conservative with increasing R in its evacuation path calculations. After a certain R value, which in this case lies between 2 and 3 m, the conservative path calculation leads to safe paths being discarded as potentially dangerous by the DES. By discarding paths that are safe at an early stage during evacuation, the DES causes a higher number of people to become trapped in the building. To further explain the effect of the value of R on the performance of DES, let us assume that a fire is present in two rooms on both sides of a corridor. However, the corridor itself is not yet affected by the hazard since the fire is still contained inside the rooms. As we increase R, we will reach a point where the corridor between the two rooms will appear to be hazardous due to the use of spatial hazard information (see Fig. 4). This results in the DES discarding every evacuation path that passes through that part of the building, limiting the number of escape paths available. This is a disadvantage in indoor environments since the number of escape paths is limited due to the spatial configuration of the building (e.g. due to physical bottlenecks, such as corridors and staircases).

The execution frequency of the distributed algorithm by the DNs also affects system performance, since the propagation of the changes in the environment depends on it. Although a high value for the algorithm execution frequency increases communication and computation costs, it also results in a more adaptive system which is able to give fast, correct suggestions to the evacuees. Simulation results indicate that the proper selection of the system parameters R and execution frequency may be non-trivial and depends on the spatial characteristics of the environment.

3.4. Simulation results: OESS

In this section, we evaluate the evacuation performance of OESS and also compare DES and OESS in the same scenario. The three-storey building used in these simulations is shown in Fig. 11. The graph that models the area consists of 240 vertices and 375 edges. The building has two exits on the first floor. All three floors are connected via staircases, with locations as shown in the figure. This is actually a model of the first three floors of the Department of Electrical and Electronic Engineering building at Imperial. The first floor is 24 m $\times$ 45 m, and other floors are 24 m $\times$ 60 m. The SN communication range is set to 5 m. Spatial hazard information is not used in the hazard metric.

In these simulations, we assume that traditional means of communication have broken down, possibly due to the hazard. Building occupants rely on the OESS for evacuation directions. We assume that CNs cannot communicate when they are located on different floors; this may be due to physical factors that affect wireless signal strength, such as thickness of the inter-floor walls. We consider two cases when OESS is used: with and without alarm. In the OESS with alarm (OESS$^+$) scenario, we assume that there is a central alarm in the building, similar to the DES and SP evacuation scenarios, and civilians start to evacuate as soon as the fire starts. In the OESS without alarm (OESS$^-$) scenario, we assume that there is no central alarm (e.g. it has failed due to power failure). Therefore, OESS provides both alerting and navigation services to building occupants. Note

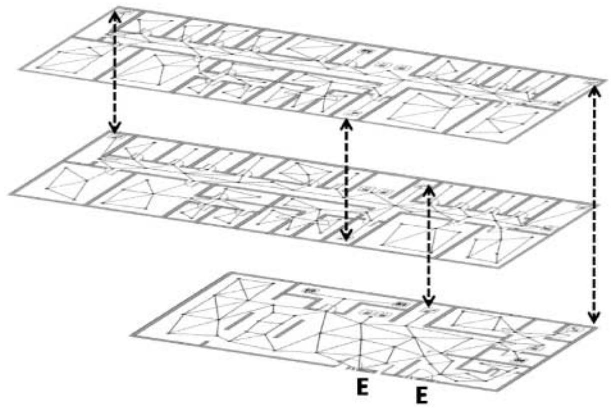

FIGURE 11. Three-floor building used in OESS and DES simulations, depicting the inter-floor connections and building exits.

TABLE 1. Message lengths in OESS simulations.

Message type	Message length (bytes)
MM	16
LM	12
Beacon	12
EM	16 (min), 52 (avg)

that CNs and SNs remain operational since they are battery powered.

All communication entities (CNs and SNs) are simulated as IEEE 802.15.4-2006 compliant devices. IEEE 802.15.4 supports both beacon-enabled and non-beacon-enabled networks, and a non-beacon-enabled implementation is more appropriate for OESS due to the mobile and *ad hoc* nature of the oppnet. In 802.15.4, non-beacon-enabled networks employ CSMA-CA[2] without RTS/CTS[3] at the MAC layer and we assume the same MAC in our simulations. The CN and SN data transfer rate is set to 100 and 20 kbits/s, respectively. IEEE 802.15.4-2006 provides raw data rates of 100 and 250 kbits/s in three PHY frequency bands. However, the actual data rate for application data will be lower due to protocol overhead and processing delays. We therefore assume an application data rate much lower than the maximum supported by 802.15.4-2006 to account for these factors. We do not explicitly simulate the PHY layer in our simulations, but we do take into account contention for the wireless medium as accessed through CSMA-CA.

In addition to the area graph and edge costs, each CN can store 100 EMs for oppcomms. Table 1 gives the message lengths used in our simulations. With 100 EMs and 52 bytes per EM on average, memory requirements for oppcomms is about 5 kB per CN.

Each data point in the presented results is an average of 50 simulation runs. Each run represents a different initial distribution of civilians, mobility patterns and hazard spread pattern (i.e. spreading rate). Where appropriate, results are presented with their 95% confidence intervals. The fire starts at the intersection of two corridors on the second floor near the staircases. This is a critical location since many escape paths pass through nearby points, meaning that many people (both on the second and third floors) will be affected soon after the hazard starts since it takes a short amount of time to affect locations close to the starting point. In addition, the hazard will spread to the first and third floors via the staircases quickly due to its proximity to the staircases.

The main purpose of OESS is safe and quick evacuation of people from the affected area through improved situational awareness. The most important evaluation criterion of evacuation performance is the number (or ratio) of people who successfully evacuate. The *evacuation ratio* is the ratio of the number of people who successfully evacuated the area to the number of all people who were in the area when the emergency started. A complementary performance parameter is *average evacuee health*, which is the average health of all people who successfully evacuated.[4] Viewed in combination with the evacuation ratio, evacuee health helps to distinguish between two outcomes where a similar number of people were evacuated but more people were exposed to the hazard in one case than the other. A third criterion is how long the evacuation process takes place, i.e. evacuation time. The *average evacuation time* is the average of the evacuation times for successfully evacuated civilians, and the *worst-case evacuation time* is the evacuation time for the last person who leaves the area.[5]

Two important spatial properties that affect oppcomms and OESS performance are population density and CN

[2]CSMA-CA: Carrier sense multiple access with collision avoidance.

[3]RTS/CTS: Request-to-send/clear-to-send mechanism employed in some IEEE MAC protocols to address the hidden node problem.

[4]Note that the average evacuee health metric does not include data from civilians that perish during evacuation.

[5]Neither average nor worst-case evacuation time metrics include data from civilians that perish during evacuation.

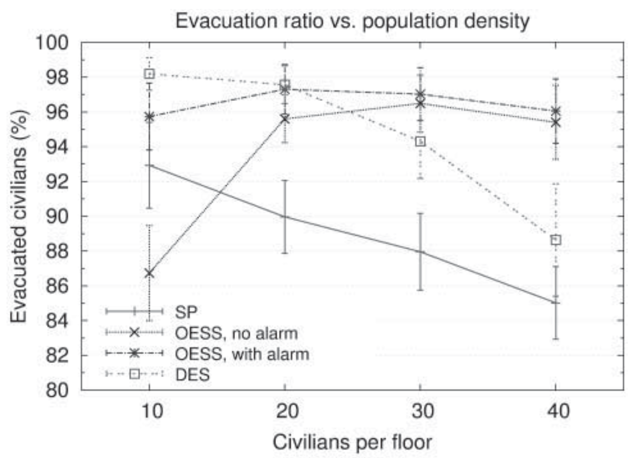

FIGURE 12. Evacuation ratio versus population density: OESS, DES and SP evacuation.

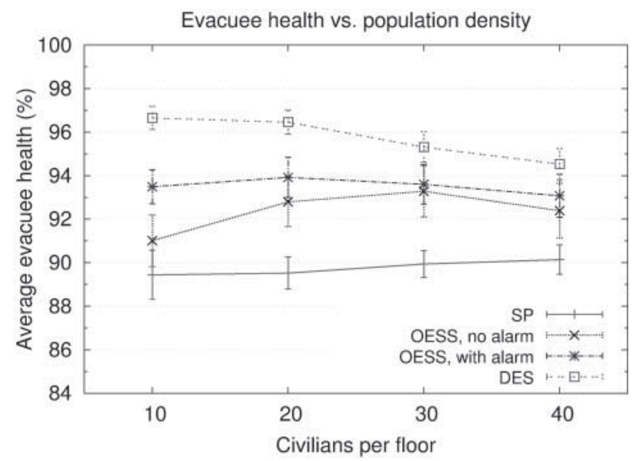

FIGURE 13. Average evacuee health versus population density: OESS, DES and SP evacuation.

communication range. Population density, or equivalently the number of people (and CNs) in the building, is directly related to the number of spatial components in OESS. We would normally expect OESS to perform better with more people in the same area since more CNs means a better connected oppnet and more contact opportunities for information exchange. We will see that this is generally true, but we will also observe that since more people need to evacuate, the evacuation will take longer and effects of physical congestion will be apparent. We will also observe that increasing the CN range improves the OESS performance due to better connectivity.

Our initial set of results evaluate the effect of population density on the evacuation performance. They also provide a comparison of OESS, DES and SP evacuation in the same settings. In DES simulations, the distributed algorithm is executed by each DN every 100 ms (i.e. 10 Hz). We assume a central alarm in the DES and SP systems. In OESS simulations, the CN communication range is set to 6 m.

Figure 12 presents the evacuation ratio versus population density for four different systems. As expected, the performance of OESS depends on the connectivity of the network and therefore on population density. As density increases, the oppnet becomes better connected and evacuation ratio is improved. This is especially apparent for OESS$^-$ since OESS provides both notification and navigation in this scenario. The effect of density is less pronounced for OESS$^+$ due to the alarm.

We observe that with 6 m communication range and 10 pf, OESS$^-$ has a lower evacuation ratio than SP, but for all other densities, OESS$^-$ performs better than SP. The SP performance decreases with increasing density. Because people do not know where the fire is in SP evacuation, they initially follow the SP (shortest in physical distance). As the number of people in the building increases, more people are affected by the fire since many of the SPs pass through hazardous areas. When OESS is in use, most of the people are guided away from the fire by their CNs based on information received via oppcomms. This means that compared with the SP case, significantly fewer people are affected by the hazard. We will also see that this has a significant effect on evacuee health when we look at Fig. 13. DES has the best evacuation ratios for low building occupancy levels (10 and 20 pf) while both OESS$^+$ and OESS$^-$ perform better than DES at higher building occupancy levels (30 and 40 pf). This indicates that DES is greatly affected by physical congestion. In contrast, OESS is less affected by congestion. For high occupancy levels (30 and 40 pf for OESS$^+$ and 40pf for OESS$^-$), we observe slightly lower evacuation ratios, suggesting that people are evacuating the building slower and therefore more are affected by the hazard. This is confirmed when we look at evacuation times in Fig. 14.

From this evaluation, we can deduce that the OESS is not suited for very sparse populations. We can address the disconnected nature of the oppnet in sparse populations by increasing the CN communication range. As we will see when we look at the effect of communication range, increasing range will increase contact opportunities and evacuation performance. However, there is a limit to communication range due to physical conditions and energy limitations. We therefore believe that while OESS is suitable for densely populated areas, a more traditional approach that is based on infrastructure wireless networks or a static node system such as DES could be used in sparser areas. We also see that OESS can tolerate alarm failure, especially in medium-to-highly populated areas.

Figure 13 shows average evacuee health versus population density. We observe that evacuee health is generally quite high, even for SP evacuation. This indicates that most of the people who successfully evacuated the building were not significantly exposed to the fire. In SP evacuation, this is mostly because the people who are exposed to fire have to backtrack and are eventually trapped inside the building, which results in them being fatally injured. Therefore they do not count in the average evacuee metric, keeping the evacuee health high. This is supported by the fact that although evacuation ratio decreases,

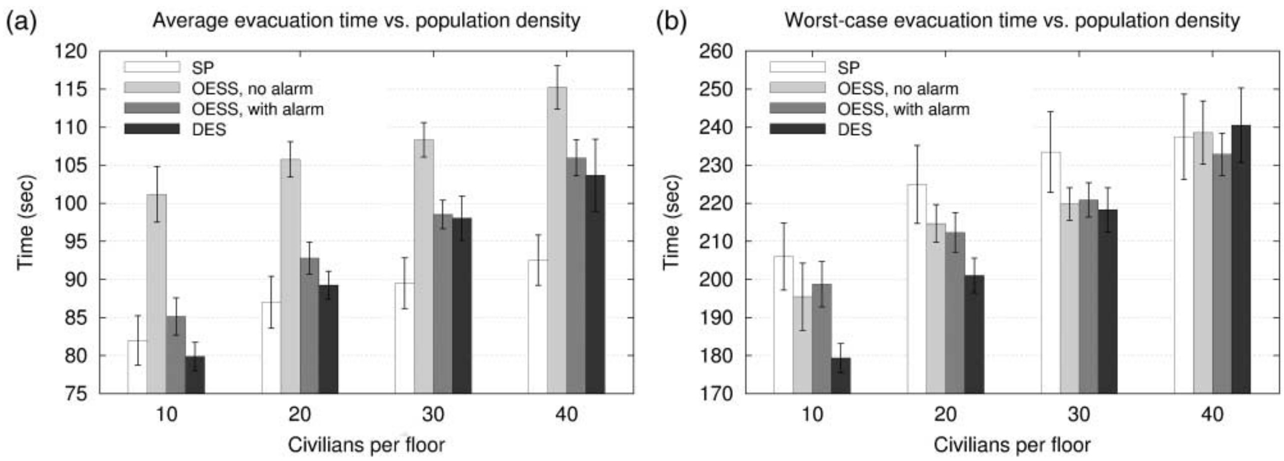

FIGURE 14. Average and worst-case evacuation time versus population density: OESS, DES and SP evacuation. (**a**) Average evacuation time. (**b**) Worst-case evacuation time.

average evacuee health does not change much as population density changes in SP evacuation.

We see that $OESS^+$ and $OESS^-$ both perform better than SP in terms of evacuee health. This is as expected since the OESS guides civilians along paths that minimize exposure to the hazard. For $OESS^-$, as population density increases, evacuee health increases in general due to better connectivity. This effect cannot be seen for $OESS^+$ due to the counter-balancing effect of physical congestion. Congestion also affects $OESS^-$ in high population densities (40 pf). Because people start to evacuate at the same time in $OESS^+$ due to the central alarm, congestion at critical locations, such as staircases, is increased. The notification of people at different times in $OESS^-$ provides a sort of natural congestion avoidance since people do not start to evacuate at the same time. Of course, as density increases, so does network connectivity and more people start evacuating at the same time, increasing congestion as observed in the 40 pf case for $OESS^-$. DES shows the best performance in evacuee health since in this case evacuees are never guided through fire. As discussed in Section 3.3, when all paths are blocked by fire, DES directs people to safe areas in the building to increase their chances of survival until emergency responders arrive. In contrast, when all paths are blocked by fire, OESS will continue to direct people towards exists along the paths that have lower cumulative hazard exposure. This behaviour can potentially increase the number of evacuated civilians at the cost of evacuee health. We observe that the performance difference between $OESS^+$ and $OESS^-$ decreases with increasing population density. We would like to note that even though the difference in evacuee health between SP and OESS is not high, when viewed together with evacuation ratio, it is apparent that OESS improves the evacuation outcome by providing intelligent and adaptive navigation services.

Our final evaluation criterion is evacuation time. This criterion is not as critical as the evacuation ratio or evacuee health but provides an idea about how fast people are evacuated in different scenarios. A faster evacuation is generally preferable since the longer the evacuation takes, the more the fire will spread and become more likely to affect civilians and block evacuation paths. Figure 14 shows the average and worst-case evacuation time versus population density. Note that casualties do not contribute to these metrics. In terms of the average evacuation time (Fig. 14a), we see that SP is generally fastest, but this is mostly due to its low evacuation ratio. What happens is that the people who do not encounter fire evacuate faster than others who have to change their paths in SP evacuation. Since many of the people who need to backtrack eventually become fatally injured in SP, the effect of their long travel time is not observed in an average evacuation time.

Although people start to evacuate the building immediately in $OESS^+$ and SP scenarios, evacuation takes on average longer in $OESS^+$ because civilians are guided along longer but safer paths with OESS compared with SP. In addition, more people successfully evacuate the building in $OESS^+$ than SP, which also increases the average evacuation time. $OESS^-$ has the longest average evacuation time since people do not start to evacuate the building as soon as the fire starts. Increasing population density increases the average evacuation time in all scenarios since evacuating more people takes longer. We also observe that the difference between $OESS^+$ and $OESS^-$ decreases with increasing density due to the diminished effect of alarm with better oppcomms connectivity. Another observation is that increasing density does not affect the average evacuation time in SP as much as it affects OESS, and this can be attributed to differences in the evacuation ratio. We see that DES has average evacuation times that are comparable with $OESS^+$ although evacuation with DES is slightly faster.

When we look at worst-case evacuation time, we see a different picture compared with average evacuation time. For example, SP is generally worse than OESS in terms of worst-case evacuation time. This is explained by the uninformed path selection and eventual backtracking involved in SP evacuation. Most of the people who take a long time during SP evacuation do so because they have to change their path one or more times due to lack of information on the hazard. Many of them become trapped in the building and die, and do not contribute to the average evacuation time. The few ones who do evacuate late do not significantly contribute to average evacuation time due to their low number and the smoothing effect of averaging. However, such evacuees do contribute very significantly to worst-case evacuation time since it is the time the last evacuee leaves the building. We therefore observe much higher worst-case evacuation times for SP compared with the average evacuation time.

As is the case for the average evacuation time, increasing density increases the worst-case evacuation time for all scenarios. We see that OESS$^-$ and OESS$^+$ show similar worst-case evacuation time performance except for the 40 pf case where OESS$^-$ takes longer. OESS$^+$ is more affected by physical congestion since people start to evacuate at the same time. On the other hand, OESS$^-$ is not affected by congestion until high densities since people start to evacuate at slightly different times. The increased effect of congestion at high density (40 pf) for OESS$^-$ is also apparent in other evacuation metrics, such as the evacuation ratio (Fig. 12), average evacuee health (Fig. 13) and average evacuation time (Fig. 14a). We see that congestion has the greatest effect on DES; DES has the best worst-case evacuation times except for the 40 pf case and worst-case evacuation time increases faster with density for DES than other systems.

In the simulations presented above, we assumed that the CN communication range is set to 6 m; this is a realistic but somewhat conservative assumption based on the expected real-life communication performance of IEEE 802.15.4-2006 compliant devices operating indoors. A longer CN range would improve the evacuation performance of OESS, especially at low densities. We discuss the effects of communication range next.

In the following simulations, we assume that there is no central alarm in the building. Figure 15 shows the evacuation ratio versus CN communication range for different population densities (10–40 pf) with OESS. We observe that increasing CN communication range increases the system performance. As can be expected, increasing CN range increases the connectivity of the CNs, which results in better dissemination of hazard information among the CNs. The improvement due to increased range decreases as population density increases. This is because with many CNs in the building, there are frequent contacts between CNs and therefore range has less effect. Population density has a more complex effect on evacuation ratio: increasing density increases evacuation ratio due to more contact opportunities, especially when the CN range is low. However, for high densities, physical congestion causes evacuation to take longer. This allows more time for the hazard to spread and affect people and decreases both evacuee health and evacuation ratio.

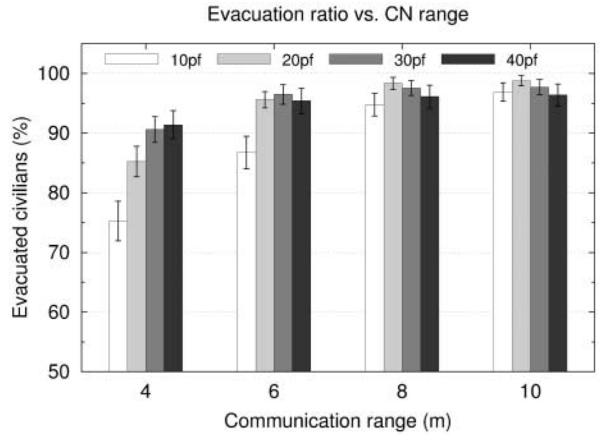

FIGURE 15. Evacuation ratio versus communication range: OESS.

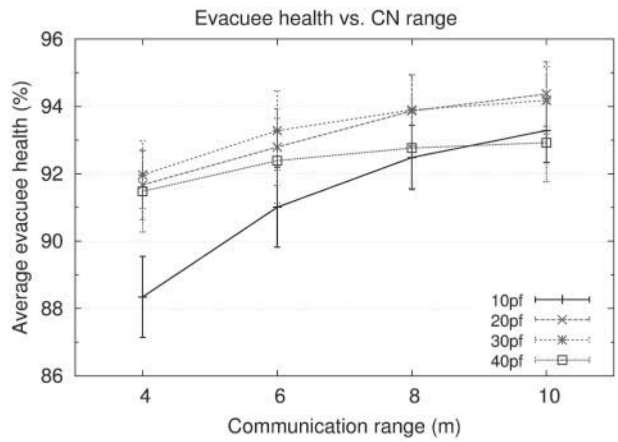

FIGURE 16. Average evacuee health versus communication range: OESS.

Figure 16 shows average evacuee health versus communication range for different population densities with OESS. We see that evacuee health follows a similar pattern to the evacuation ratio as range and density increase. However, average evacuee health is generally quite high and the differences in health are much less compared with differences in evacuation ratio, indicating that OESS directs people via safe paths to exits. Most of the deaths are due to late notification of people. CNs that are disconnected from the oppnet for long times do not get the EMs from other CNs and therefore do not know that there is an emergency in the building. This causes their users to start evacuation late and these people have a much higher probability of becoming casualties since the hazard spreads further and blocks more evacuation paths.

Figure 17 shows average and worst-case evacuation time versus communication range for different population densities with OESS. For both metrics, two main properties are

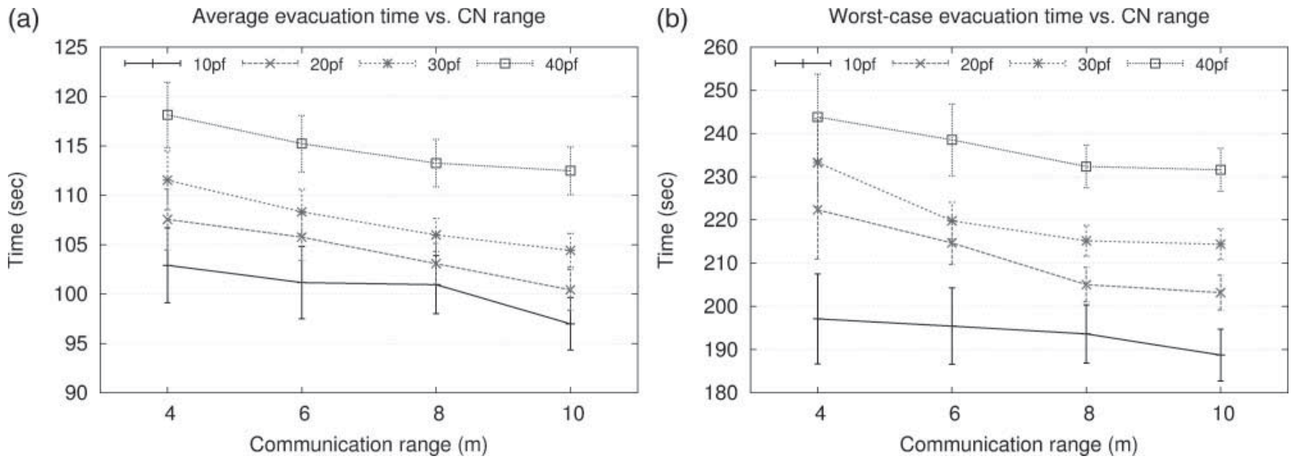

FIGURE 17. Average and worst-case evacuation time versus communication range: OESS. (**a**) Average evacuation time. (**b**) Worst-case evacuation time.

apparent: increasing CN range decreases evacuation time, and increasing density increases the evacuation time. Increasing range increases oppnet connectivity and decreases the number of people who receive late notifications. Furthermore, with increased range CNs receive hazard measurements from farther locations faster and therefore CNs have a more up-to-date view of the hazard. This translates into better routing of people since backtracking due to incorrect information is reduced. All these factors decrease the evacuation time. The effect of population density on evacuation time is as expected. The more people there are in the building, the longer it takes to evacuate them. In addition, congestion effects start to become more important with higher densities due to the bottlenecks in the building, such as staircases. It is good to note that although evacuation takes longer with higher densities, the evacuation ratio is not affected that much, indicating that OESS can adapt evacuation routes and direct evacuees safely along the fastest routes.

Our results with different CN communication ranges indicate that a higher range generally improves the evacuation outcome. However, the effect of increased range diminishes as population density increases, as discussed above. Population density is a property of the environment and therefore cannot be controlled by the system. Range, on the other hand, can be controlled within certain limits. For example, by increasing its transmission power, a CN can increase its range while also increasing its energy use. Note that there is a limit on maximum transmission power and therefore maximum range. We believe that dynamic adjustment of range individually by each CN based on its 'sensed' population density is an appropriate approach to maximize the utility of range. The scheme would work as follows: a CN would start out with its maximum range and continually sense how many neighbours it has. When the number of neighbours is high, it will decrease its range to the next level. If the number of neighbours is still high, it will further decrease its range, etc. When number of neighbours is low, the range is increased in levels, up to the maximum possible. The range levels and corresponding numbers of neighbours depend on spatial properties of the building, and these values need to be set properly for effective operation. We leave these as future work.

3.5. Remarks

We have presented two different spatially distributed computing systems to provide emergency evacuation support in indoor environments. DES is mainly based on nodes (DNs and SNs) that are pre-deployed in the area, whereas OESS is a hybrid system consisting of mobile (CNs) and static (SNs) components. The two systems can be deployed either as alternatives or complementary to each other. DES is a more traditional system and it is attractive since the operation of the system is independent of the number of people and of their movement. Therefore, DES provides good evacuation performance even when the number of people in the building is small. OESS, on the other hand, exploits the movement of people and is dependent on the number of people in the area. As discussed in Section 3.4, OESS does not perform well when the number of people is small. For buildings where the number of occupants is small or highly variable, DES is a more suitable option. For all other cases, OESS is preferable since it provides equivalent or better evacuation performance than DES, and has additional desirable properties.

Oppcomms provides disruption-tolerant communications in OESS and the mobile nature of CNs render OESS highly resilient to failures. DN failures can greatly impact the performance of DES, while OESS is much more resilient to CN and SN failures. To handle failures in DES, OESS can be used as a backup system as described in [27].

DES is targeted for indoor environments and it would be difficult to deploy DES in large-scale outdoor urban environments. OESS is more suitable for large-scale outdoor deployment once we remove the need for SNs. This can be achieved by integrating the sensors into CNs, and using a system like GPS for localization.

An advantage OESS has over DES is the potential use of consumer devices, such as smartphones, already owned and carried by people as CNs. The use of such devices would increase the availability of OESS and decrease deployment costs.

We observed that physical congestion adversely affects the evacuation outcome, independent of whether DES or OESS was used, although our evaluation indicates that DES is more sensitive to congestion. A solution would be to modify the definion of the 'effective length' in order to derive a metric that also takes congestion into account. With DES, a mechanism to detect people is required, either via wireless communications if every person is equipped with a mobile device or via cameras in the building. OESS already supports detection via wireless communication since every person has a CN. Our preliminary work on congestion avoidance has shown that physical congestion is unavoidable in many indoor environments due to their spatial properties, and that congestion avoidance becomes effective when population density is quite high ($\geq$50 people per floor for the building given in Fig. 11). With lower densities, the extra time required to take a longer path to avoid congested paths is usually higher than the time lost due to congestion. We leave further investigation of congestion avoidance as future work.

The building layout, represented as a graph, is implicitly required in DES since SNs and DNs are deployed according to the graph. However, each DN does not need to know the whole building graph. In the OESS, each CN knows the whole building graph and we believe that this is a reasonable assumption since the building graph does not change except for hazard intensities, which are kept separate from the graph. CNs can obtain the graph off-line or via wireless communication with SNs as they enter the building. If the building graph is unavailable or unknown, then anchor points with known locations in the building, such as SNs, can be used by CNs to create a (partial) map of the building prior to the emergency. A further possibility is the dissemination of partial graph information via oppcomms. This possibility is explored in [28], discussed in the next section.

4. RELATED WORK

There are various approaches regarding the problem of navigation and evacuation during emergencies. In [29], the authors propose a distributed algorithm for robot navigation using a sensor network. They evaluate their approach using a robot and a sensor network composed of nine nodes. They do not, however, take into account other parameters such as a dynamically spreading hazard or multiple evacuees, both of which are important characteristics of an urban emergency. The authors in [30] propose an algorithm inspired by sensor network routing in order to guide a flying robot. Although they also evaluate their method for guiding humans, their evaluation scenario includes only one person and 12 sensors positioned in a building. Scenarios with more occupants and dynamically changing conditions are not investigated. In [31] a system based on sensor networks is proposed, for navigating the user to a goal location by avoiding hazardous areas. The path calculation algorithm is based on artificial potential fields. A testbed of 50 wireless nodes is used to evaluate the approach. The focus of the system is on the time needed by the nodes to obtain the SP. The authors, however, do not include an evaluation scenario with a spreading hazard or a large number of people.

A distributed navigation algorithm geared towards emergency situations is presented in [32]. The approach is inspired by an *ad hoc* network routing protocol and uses hop-count as the distance metric. The authors test their method using simulations in various network topologies and hazard locations. Although the algorithm is able to find exit paths avoiding hazardous areas, the hazard is static and the presence of evacuees inside the area is not taken into account. As such, this work focuses on the calculation of SPs that avoid certain static areas. In [33], the work in [32] is extended to 3D indoor building environments by adding stairway nodes to connect different floors, while in [34] a distributed WSN-based building evacuation algorithm that takes into account the expected spread of the hazard inside the building is discussed. In [35, 36], congestion during evacuation is taken into account; in [35], how to guide rescuers to critical areas in the building is also examined. Other work [37] presents an autonomous indoor navigation system based on mobile phones that assumes a WSN to monitor the building and receive updates on a dynamic hazard, with a centralized emergency guidance system.

The aforementioned approaches generally focus on how efficiently a (sensor) network can find a path to a location of interest. Although the notion of hazard is present in most of them, the context in which each system functions is not directly related to an emergency situation. A common assumption is a static hazard representation and the absence of evacuees that interact with the system. Our proposed systems, however, provide adaptive directions in a dynamic (spreading) hazard, and we consider all stages of emergency evacuation, from notification of building occupants to navigation of many evacuees. Another difference is that our systems keep the decision making and sensing components separate, and SNs do not form a WSN to preserve energy. DES is similar in concept to the works mentioned above due to its static infrastructure, while OESS is considerably different due to the use of mobile nodes and oppcomms.

Works discussed above generally derive from research on robot navigation and WSNs. There are also various approaches from the spatial computing domain closely related to the problem of evacuation and emergency support. An approach that uses spatial computers as the building block for the design of distributed algorithms is proposed in [38]. It is based on two observations: (i) a device communicates only with neighbours over short distances, and (ii) the overall structure of the network forms a discrete approximation of the structure of the space of interest. The authors use the abstraction of amorphous medium to geometrically formulate distributed applications and give an example of their modelling approach in the context of tank tracking. Finally, they indicate how the use of this geometric abstraction can benefit the adaptation capability of distributed algorithms. This work is of relevance in that DES is based on concepts that closely follow the two observations above.

A distributed path calculation algorithm based on self-healing gradients is presented in [39]. In a fixed network of wireless devices that communicate over unreliable links and where nodes are subject to failures, the algorithm self-stabilizes in O(diameter) time. The algorithm is evaluated using simulation and real WSN-testbed experiments, where it is used to calculate minimum-threat paths to a destination in the presence of a threat. DES employs a similar distributed approach to path calculation, but it provides adaptive hop-by-hop paths to multiple destinations in the presence of a dynamic threat.

The idea of using peer-to-peer communications between nearby mobile clients to collaboratively share data is proposed in [40]. Mobile clients are assumed to issue queries regarding their geographic surroundings to a central server. Since adjacent mobile devices are likely to issue queries with overlapping results, it is more efficient for such devices to share data. A grid-based structure is used to manage the data distributed among devices. Instead of using flooding for searching among peers, routing tables are maintained, which increase the search speed and reduce communication overhead. Using the proposed scheme to answer a location-based spatial query, a device first checks its local storage. If data are not available locally, they request the data from the corresponding peer using the routing table. Communication with a central server is used only when data cannot be obtained using the aforementioned ways. The approach is evaluated using a Java simulation framework. The size of the spatial grid used is 1000×1000 points. The authors investigate the effect of parameters such as device storage size, spatial grid configuration and update ratios on the performance of the system. Their results indicate that their proposed approach is efficient in answering queries and robust to data updates. This work supports that short-range local communications among mobile devices, as employed in OESS, can be effective to share information.

Winter *et al.* propose the use of short-range communications between mobile entities to enable collaborative evacuation based on local knowledge in [28]. The emergency area is represented as a graph and they consider three scenarios regarding knowledge of the area: (i) complete knowledge, (ii) partial knowledge, obtained prior to the emergency as the entity moves in the area and (iii) no knowledge. The static hazard location is not known beforehand in any scenario and a central alarm is assumed. The authors then evaluate the utility of sharing both hazard and graph information among entities as they come in contact during evacuation using simulation experiments of three different graphs: densely connected, sparsely connected (similar to a road network) and single building floor. The authors find that in most scenarios, sharing information improves evacuation and when a dynamic hazard is considered, the benefit of communication is even greater. This work supports that sharing graph and hazard information via short-range communications among mobile entities improves evacuation. While valuable, this work focuses on the concept of local sharing of knowledge to improve evacuation but does not provide any solutions to practical problems, such as system design, indoor localization and real-time monitoring of the hazard.

A spatial orientation and information system for indoor spatial awareness is proposed in [41]. The authors propose a system composed of an inertial measurement unit (IMU), a stereo camera system and a step sensor for tracking the movements of the user inside the building. The IMU is used for indoor localization, while the camera is used for heading estimation and the step sensor for detecting movement. It is assumed that the initial location of the user is known. The system can then track the movements of the user without using wireless communication or magnetic field sensing. DES does not need indoor localization, and OESS uses wireless communication between CNs and anchor points with known locations (SNs) for CN localization. The system proposed in [41] provides an alternative indoor localization solution for an OESS design where SNs are made redundant by embedding the sensors into CNs. Although feasible, we believe that the localization approach proposed in [41] is unpractical with current technology due to many additional sensor requirements and the difficulty of integrating these into a small form factor communication device.

5. CONCLUSIONS AND FUTURE WORK

We have proposed two spatially distributed computing systems that provide intelligent navigation services for evacuation support during an emergency. The first system (DES) is based on static DNs that run a distributed algorithm. The DES operates as a distributed computer that adapts to changing conditions. The 'best' evacuation paths are calculated collectively by the DNs using local information and communication. The second system (OESS) comprises wireless CNs that are carried by the civilians. The CNs form an oppnet which enables the exchange of EMs for alerting and guiding civilians during evacuation. The OESS can be viewed as a hybrid spatial computer where parts of the system are mobile (CNs) and others are static (SNs). Both systems are supported by pre-deployed sensors (SNs) that provide real-time information on the hazard. We evaluate our systems using a distributed simulation platform (DBES). Our simulation results show that the presence of the evacuation systems benefits the evacuation procedure. Our simulation study also shows how parameters such as communication range, execution and communication frequency and population density, affect the system performance.

In future work we will study the performance of our systems when failures are present, as well as mechanisms which can improve their performance under such conditions. We also plan on investigating security issues of oppcomms for emergency support, specifically looking at the effect of malicious users in the system and how we can defend against them. Integrating congestion avoidance mechanisms into our systems is another area of future work.

We have developed our systems for evacuation support in indoor areas, such as buildings. These systems, especially OESS, could easily be deployed in large-scale urban areas. We believe that evaluation in such settings can provide important insights into the effect of spatial properties of an area on the evacuation performance.

As we look at areas for future research, we think that mathematical modelling of emergency management systems has not been sufficiently explored, while it is a standard technique in computer system performance evaluation [42], as well as in distributed systems [43]. Another useful approach would be to evaluate the effect of uncertainty in the sources of information [44] and in their interpretation. An advantage of such approaches is that they can provide computationally fast mathematical prediction, much faster than the simulation methods described in this paper, so that the outcome of a large number of simulations can be replaced by the solution of a probability model. We think that such methods are worth investigating in view of the increasing importance of emergency management and more generally for the study of human-based cyber-technical systems.

FUNDING

This work was supported by the UK Technology Strategy Board under the SATURN (Self-organizing Adaptive Technology underlying Resilient Networks) research project.

REFERENCES

[1] Shields, T.J. and Boyce, K.E. (2000) A study of evacuation from large retail stores. *Fire Safety J.*, **35**, 25–49.
[2] Pauls, J. (1999) A personal perspective on research, consulting and codes/standards development in fire-related human behaviour, 1969–1999, with an emphasis on space and time factors. *Fire Mater.*, **23**, 265–272.
[3] MEMSIC Inc. (2012) TelosB wireless module datasheet. http://www.memsic.com/support/documentation/wireless-sensor-networks/category/7-datasheets.html. 6020-0094-04 Rev B.
[4] MEMSIC Inc. (2012) MicaZ wireless module datasheet. http://www.memsic.com/support/documentation/wireless-sensor-networks/category/7-datasheets.html. 6020-0065-05 Rev A.
[5] Elechouse. (2012) Smoke sensor module. http://www.elechouse.com/elechouse/index.php?main_page=product_info&cPath=152_231&products_id=710.
[6] Parallax Inc. (2010) CO (carbon monoxide) gas sensor module (no.27931) documentation. http://www.parallax.com/Portals/0/Downloads/docs/prod/sens/27904_27930-1-2-GasSensor-v2.2.pdf. v2.2, 20100802.
[7] Parallax Inc. (2010) CO_2 (carbon dioxide) gas sensor module (no.27929) documentation. http://www.parallax.com/Portals/0/Downloads/docs/prod/sens/27929-GasSensor-v1.0.pdf. v1.0, 20100225.
[8] Filippoupolitis, A. and Gelenbe, E. (2009) A Distributed Decision Support System for Building Evacuation. *Proc. 2nd IEEE Int. Conf. on Human System Interaction*, Catania, Italy, May 21–23, pp. 323–330. IEEE, New York, NY, USA.
[9] Elms, D., Buchanan, A. and Dusing, J. (1984) Modeling fire spread in buildings. *Fire Technol.*, **20**, 11–19.
[10] Hasofer, A. and Odigie, D. (2001) Stochastic modelling for occupant safety in a building fire. *Fire Safety J.*, **36**, 269–289.
[11] Gelenbe, E., Seref, E. and Xu, Z. (2001) Simulation with learning agents. *Proc. IEEE*, **89**, 148–157.
[12] Gelenbe, E. (2004) Sensible decisions based on QoS. *Comput. Manage. Sci.*, **1**, 1–14.
[13] Humblet, P.A. (1991) Another adaptive distributed shortest path algorithm. *IEEE Trans. Commun.*, **39**, 995–1003.
[14] Gelenbe, E., Lent, R. and Xu, Z. (2001) Measurement and performance of a cognitive packet network. *Comput. Netw.*, **37**, 691–701.
[15] Schneider, M. (1993) Self-stabilization. *ACM Comput. Surv.*, **25**, 45–67.
[16] Gorbil, G. and Gelenbe, E. (2011) Opportunistic communications for emergency support systems. *Procedia Comput. Sci.*, **5**, 39–47.
[17] Pelusi, L., Passarella, A. and Conti, M. (2006) Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. *IEEE Commun. Mag.*, **44**, 134–141.
[18] Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C. (2001) *Introduction to Algorithms*. MIT Press and McGraw-Hill, Cambridge, MA, USA.
[19] Vahdat, A. and Becker, D. (2000) Epidemic Routing for Partially-Connected Ad Hoc Networks. Technical Report CS-2000-06, Duke University, Department of Computer Science, Durham, NC, USA.
[20] Song, L. and Kotz, D.F. (2007) Evaluating Opportunistic Routing Protocols with Large Realistic Contact Traces. *Proc. 2nd ACM Workshop on Challenged Networks*, Montreal, QC, Canada, September 14, pp. 35–42. ACM, New York, NY, USA.
[21] Dimakis, N., Filippoupolitis, A. and Gelenbe, E. (2010) Distributed building evacuation simulator for smart emergency management. *Comput. J.*, **53**, 1384–1400.
[22] Bellifemine, F.L., Caire, G. and Greenwood, D. (2007) *Developing Multi-Agent Systems with JADE*. John Wiley and Sons Ltd, Chichester, UK.
[23] Filippoupolitis, A., Hey, L., Loukas, G., Gelenbe, E. and Timotheou, S. (2008) Emergency Response Simulation Using Wireless Sensor Networks. *Proc. 1st Int. Conf. on Ambient Media and Systems*, Quebec City, Canada, February 11–14, pp. 1–8. ICST, Brussels, Belgium.
[24] Timotheou, S. and Loukas, G. (2009) Autonomous Networked Robots for the Establishment of Wireless Communication in Uncertain Emergency Response Scenarios. *Proc. 24th ACM Symp. on Applied Computing, Track on Intelligent Robotic Systems*, Honolulu, HI, USA, Mar 8–12, pp. 1171–1175. ACM, New York, NY, USA.

[25] Loukas, G. and Timotheou, S. (2008) Connecting Trapped Civilians to an ad hoc Network of Emergency Response Robots. *Proc. 11th IEEE Int. Conf. on Communications Systems (ICCS'08)*, Guangzhou, China, November 19–21, pp. 599–603. IEEE, New York, NY, USA.

[26] Loukas, G., Timotheou, S. and Gelenbe, E. (2008) Robotic Wireless Network Connection of Civilians for Emergency Response Operations. *Proc. 23rd Int. Symp. on Computer and Information Sciences (ISCIS'08)*, Istanbul, Turkey, October 27–29, pp. 1–6. IEEE, New York, NY, USA.

[27] Filippoupolitis, A., Gorbil, G. and Gelenbe, E. (2011) Autonomous Navigation Systems for Emergency Management in Buildings. *Proc. 54th Annual IEEE Global Telecommunications Conf. (GLOBECOM'11) Workshops*, Houston, TX, USA, December 5–9, pp. 1096–1101. IEEE, New York, NY, USA.

[28] Winter, S., Richter, K.-F., Shi, M. and Gan, H.-S. (2011) Get Me Out of Here: Collaborative Evacuation Based on Local Knowledge. *Proc. 3rd ACM SIGSPATIAL Int. Workshop on Indoor Spatial Awareness*, Chicago, IL, USA, November 1, pp. 35–42. ACM, New York, NY, USA.

[29] Batalin, M., Sukhatme, G.S. and Hattig, M. (2004) Mobile Robot Navigation Using a Sensor Network. *IEEE Int. Conf. on Robotics and Automation*, New Orleans, LA, USA, April 26–May 1, pp. 636–642. IEEE Press, Piscataway, NJ, USA.

[30] Corke, P., Peterson, R. and Rus, D. (2003) Networked Robots: Flying Robot Navigation Using a Sensor Net. *11th Int. Symp. of Robotics Research (ISRR'03)*, Siena, Italy, October 19–22, pp. 234–243. Springer Berlin/Heidelberg, Berlin, Germany.

[31] Li, Q., Rosa, M.D. and Rus, D. (2003) Distributed Algorithms for Guiding Navigation Across a Sensor Network. *Proc. 9th Annual Int. Conf. on Mobile Computing and Networking (MobiCom'03)*, San Diego, CA, USA, September 14–19, pp. 313–325. ACM, New York, NY, USA.

[32] Tseng, Y., Pan, M. and Tsai, Y. (2006) Wireless sensor networks for emergency navigation. *Computer*, **39**, 55–62.

[33] Pan, M., Tsai, C. and Tseng, Y. (2006) Emergency guiding and monitoring applications in indoor 3D environments by wireless sensor networks. *Int. J. Sensor Netw. (IJSNet)*, **1**, 2–10.

[34] Barnes, M., Leather, H. and Arvind, D.K. (2007) Emergency Evacuation Using Wireless Sensor Networks. *Proc. 32nd IEEE Conf. on Local Computer Networks (LCN'07)*, Dublin, Ireland, October 15–18, pp. 851–857. IEEE Computer Society, Los Alamitos, CA, USA.

[35] Li, S., Zhan, A., Wu, X. and Chen, G. (2009) ERN: Emergency Rescue Navigation with Wireless Sensor Networks. *Proc. 15th Int. Conf. on Parallel and Distributed Systems (ICPADS'09)*, Shenzhen, China, December 8–11, pp. 361–368. IEEE Computer Society, Los Alamitos, CA, USA.

[36] van Willigen, W., Neef, R., van Lieburg, A. and Schut, M.C. (2009) WILLEM: A Wireless Intelligent Evacuation Method. *Proc. 3rd Int. Conf. on Sensor Technologies and Applications (SENSORCOMM'09)*, Glyfada, Greece, June 18–23, pp. 382–387. IEEE Computer Society, Los Alamitos, CA, USA.

[37] Inoue, Y., Sashima, A., Ikeda, T. and Kurumatani, K. (2008) Indoor Emergency Evacuation Service on Autonomous Navigation System Using Mobile Phone. *Proc. 2nd Int. Symp. on Universal Communication (ISUC'08)*, Osaka, Japan, December 15–16, pp. 79–85. IEEE Computer Society, Los Alamitos, CA, USA.

[38] Beal, J. and Schantz, R. (2010) A Spatial Computing Approach to Distributed Algorithms. *Proc. 45th Asilomar Conf. on Signals, Systems and Computers*, Pacific Grove, CA, USA, November 7–10, pp. 1–5. IEEE Signal Processing Society Press, Piscataway, NJ, USA.

[39] Beal, J., Bachrach, J., Vickery, D. and Tobenkin, M. (2008) Fast Self-Healing Gradients. *Proc. 2008 ACM Symp. on Applied Computing*, Fortaleza, Ceara, Brazil, March 16–20, pp. 1969–1975. ACM, New York, NY, USA.

[40] Huang, Z., Jensen, C., Lu, H. and Ooi, B. (2007) Collaborative Spatial Data Sharing Among Mobile Lightweight Devices. In Papadias, D., Zhang, D. and Kollios, G. (eds), *Advances in Spatial and Temporal Databases*. Lecture Notes in Computer Science 4605, pp. 366–384. Springer Berlin/Heidelberg, Berlin, Germany.

[41] Li, R., Skopljak, B., He, S., Tang, P., Yilmaz, A. and Jiang, J. (2010) A Spatial Orientation and Information System for Indoor Spatial Awareness. *Proc. 2nd ACM SIGSPATIAL Int. Workshop on Indoor Spatial Awareness*, San Jose, CA, USA, November 2, pp. 10–15. ACM, New York, NY, USA.

[42] Gelenbe, E. (1973) A unified approach to the evaluation of a class of replacement algorithms. *IEEE Trans. Comput.*, **C-22**, 611–618.

[43] Gelenbe, E. and Sevcik, K.C. (1979) Analysis of update synchronisation algorithms for multiple copy data bases. *IEEE Trans. Comput.*, **C-28**, 737–747.

[44] Gelenbe, E. and Hebrail, G. (1986) A Probability Model of Uncertainty in Data Bases. *Proc. 2nd Int. Conf. on Data Engineering (ICDE'86)*, Los Angeles, CA, USA, February 5–6, pp. 328–333. IEEE Computer Society, Washington, DC, USA.