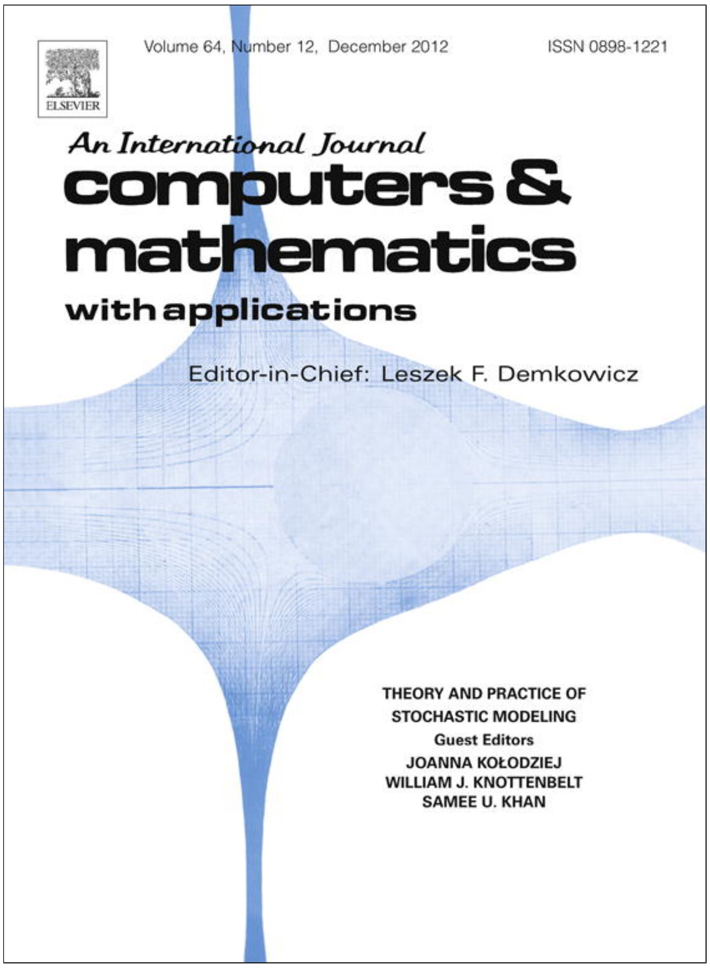
Volume 64, Number 12, December 2012
ISSN 0898-1221
ELSEVIER
An International Journal
computers & mathematics
with applications
Editor-in-Chief: Leszek F. Demkowicz
THEORY AND PRACTICE OF
STOCHASTIC MODELING
Guest Editors
JOANNA KOŁODZIEJ
WILLIAM J. KNOTTENBELT
SAMEE U. KHAN

Computers and Mathematics with Applications 64 (2012) 3869–3880

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Large scale simulation for human evacuation and rescue

Erol Gelenbe *, Fang-Jing Wu [1]

Intelligent Systems and Networks Group, Department of Electrical & Electronic Engineering, Imperial College, London SW7 2AZ, UK

ARTICLE INFO

Keywords:
Cyber-physical systems
Emergency management
Search and rescue systems
Distributed decision making
Modeling and simulation

ABSTRACT

This paper surveys recent research on the use of sensor networks, communications and computer systems to enhance the human outcome of emergency situations. Areas covered include sensing, communication with evacuees and emergency personnel, path finding algorithms for safe evacuation, simulation and prediction, and decision tools. The systems being considered are a special instance of real-time cyber-physical-human systems that have become a crucial component of all large scale physical infrastructures such as buildings, campuses, sports and entertainment venues, and transportation hubs.

1. Introduction and overall vision

Cyber-technical systems (CPS) that exploit wireless technologies, micro-sensing micro-electro-mechanical-systems (MEMS), and distributed decision making and control, have enriched the confluence of ubiquitous computing, networking technologies, and wireless sensor networks (WSNs), boosting many promising applications in environmental sensing [1], health monitoring [2], surveillance [3], intelligent transportation [4], guiding groups on tourist tours [5], and emergency response [6,7]. In particular this paper focuses on sensor-aided CPSs that enable intelligent and fast response to emergencies such as fires, earthquakes, or terrorist attacks. Real-time monitoring and quick response are inherent requirements in the design of an emergency response system. As an example, during a fire many different types of sensors can cooperate to interact with civilians and the environment. Temperature and gas sensors are responsible for monitoring the spreading of hazards. Rotatable cameras track the spread of the fire and the movement of civilians. Ultrasonic sensors can range the distance to obstacles in the environment, and monitor dynamic changes of maps due to the sudden changes of some built structures through destruction and the accumulation of debris. Intelligent evacuation scheduling can be conducted by the cooperation between first-aid decision nodes, sensors, and civilians with mobile devices since partial information and opportunistic connection are usually inevitable in an emergency. Civilians with mobile devices will follow personalized navigation directions and distributed decisions may help mitigate congestion, while those without mobile devices may follow audio or visible LED directions from nodes in their neighborhood. Grid/Cloud-supported simulators will gather all sensing information to dynamically predict and forecast the spread of hazards and to make decisions on resource allocation and response policies.

1.1. Approaches to emergency response

Two types of approaches to emergency response, Approach 1 and Approach 2 of Fig. 1, have motivated considerable research. One approach addresses the evacuation of victims of an emergency with the aid of fixed wireless sensor networks, so that the evacuation process responds dynamically to the manner in which hazards spread or recede. In this approach, work has also been devoted to directing first responders and emergency personnel towards the events that are taking place

* Corresponding author.
E-mail addresses: e.gelenbe@imperial.ac.uk (E. Gelenbe), f.wu@ntu.edu.sg (F.-J. Wu).

[1] Present address: Nanyang Technological University, Singapore.

0898-1221/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2012.03.056

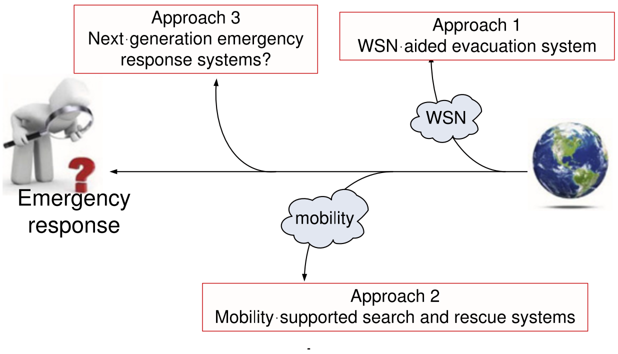

Fig. 1. Solutions to emergency disaster responses.

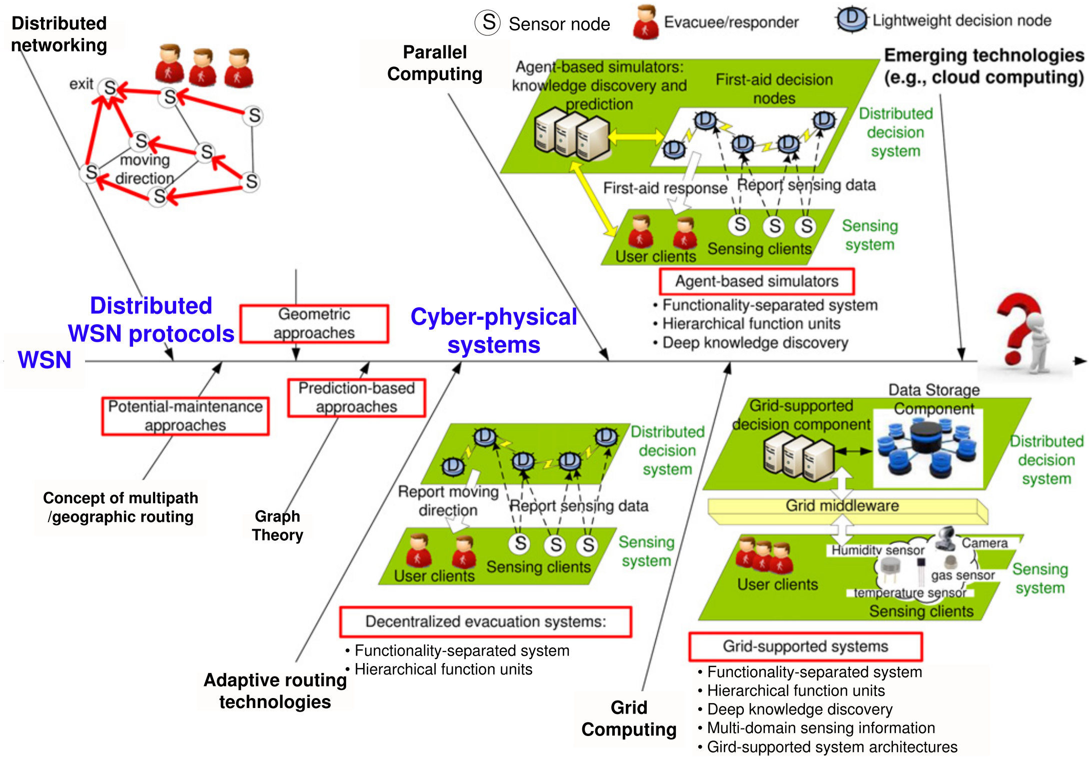

Fig. 2. Analysis of emergency evacuation systems.

and to aid the victims. Approach 2 focuses on the use of mobile devices, as well as sensors, so that the victims of an emergency and the emergency personnel can act autonomously based on the advice and information that they receive. Approach 3 on the other hand which we have called "next generation systems" would include some of the methods and technologies that we will discuss in Section 5.

2. Emergency evacuation

A representation of the evolution of emergency evacuation systems, from the simpler to the more complex, is given in Fig. 2. Below, we review both types of systems and discuss issues including communications, information acquisition and dissemination, knowledge discovery, heterogeneous system integration and asynchronous control.

Finding safe evacuation paths and providing them in a timely fashion to the evacuees is the primary goal of an emergency evacuation system. We will therefore discuss three classes of distributed protocols to compute safest evacuation paths.

2.1. Potential-maintenance approaches

In [8], a WSN is adopted to monitor hazards in the environment, and only one exit is assumed. Each user is equipped with a sensor node to communicate with the WSN for requesting an emergency evacuation path to the exit. The deployment of the WSN is assumed to be known by each user. The sensors that detect hazards are modeled as multiple obstacles. Thus, the goal is to find a "safest" path from each sensor to the exit without passing through any obstacles. The concept of *artificial* *potential fields* which has been long used in mission planning [9,10] is adopted to compute evacuation paths in a distributed manner. The exit sensor generates an attractive potential to pull sensors to the exit, while each obstacle generates a repulsive potential to push sensors away from the obstacle. In this way, each sensor can compute an overall potential value that is used to guide the evacuees. Here the overall potential value of sensor s_i is computed by $P(s_i) = \sum_{s_h \in H} \frac{1}{d^2(s_i, s_h)}$, where H is the set of sensors that detect hazard and $d(s_i, s_h)$ is the shortest hop distance from s_i to an obstacle s_h (i.e., hazardous sensor).

However, as only the shortest path is used without the concept of *hazardous regions*, [8], paths that are used may be very close to the sources of hazards. Also, wireless links do not offer accurate navigation links so that an impractical path passing through physical obstacles (e.g., walls) may be provided. Thus, by considering several hazardous regions, each of which is formed by a set of sensors whose hop distance from a hazard is not greater than a predefined threshold D, and a manual *navigation graph*, in [11] the concept of *multipath routing* in mobile ad hoc networks navigate people as far away from hazardous regions as possible. Each sensor node will maintain an *altitude* to guide people to the neighboring sensor node in the navigation graph with the lowest altitude. To bypass hazardous regions, sensor nodes in hazardous regions must raise their altitudes by the following way. When sensor s_i is informed by a hazardous sensor s_h with $d(s_i, s_h) \leq D$, it will consider itself within the hazardous region and update its altitude by $A'(s_i) = \max\{A(s_i), A_{emg} \times \frac{1}{d^2(s_i, s_h)} + d(s_i, s_e)\}$, where A_{emg} is a large constant for the altitude of the sensors that detect hazards and $d(s_i, s_e)$ is the shortest hop distance from s_i to an exit. Here, $A'(s_i)$ and $A(s_i)$ are used to distinguish the altitude before from after update. Since s_i may be within multiple hazardous regions, the maximum altitude resulted by these hazardous regions accounts for s_i's altitude. By setting the altitude, some sensors may become local minimum ones. A partial link reversal operation is performed to solve this problem so that each sensor maintains at least one outgoing link. Ref. [12] extends [11] to a 3D environment, where sensors are classified as normal sensors, exit sensors, and stair sensors. A sensor considers itself in a hazardous region if it is within D hops away from hazards, or if it is a stair sensor and its downstairs sensors are in hazardous regions. The navigation principle is to guide people to rooftops if there are no safe paths to "downstairs".

However, frequent global message flooding should be avoided when hazards dynamically expand or shrink. Thus [13] exploits *localized geographic routing* to plan navigation paths so as to adapt to dynamic hazards, where only those sensors without outgoing links need to perform a local link reversal operation in case of changes in hazardous regions. Each sensor is assumed to know its geographic coordinates. Not only hazardous regions but also *safe regions* are considered. A hazardous region is defined as the area whose danger degree exceeds a predefined threshold, while the a safe region is defined as the area outside those hazardous regions. The goal is to find at least one *safe path* for each sensor in the safe regions and at least one *escape path* for each sensor in the hazardous regions. Each sensor s_i in the safe region will maintain a safe vector (R_i, d_i, s_i), where R_i is the reversal counter of s_i in safe regions which indicates when s_i becomes a local minimum, and d_i is the Euclidean distance to the nearest exit. In contrast, each sensor s_i in hazardous regions maintains a hazardous vector $(\overline{R_i}, \overline{d_i}, s_i)$, where $\overline{R_i}$ is the reversal counter of s_i in hazardous regions which indicates the times of s_i becoming a local maximum and $\overline{d_i}$ is the Euclidean distance to the hazard source. Based on the safe vector or the hazard vector, each sensor can set a navigation link for moving toward exits or for escaping from hazardous regions. If there exists a local minimum/maximum at s_i, only those incoming links from neighboring sensors whose reversal counter is less than s_i' will be reversed so that each sensor node has at least one outgoing link for navigation. The partial reversal operation will efficiently reduce communication overhead when hazards change dynamically.

The human *congestion problem* in an emergency evacuation is considered in [14,15], where [14] proposes a distributed protocol to balance the number of evacuees among multiple navigation paths to different exits. In this work, each sensor is assumed to be location aware and capable of detecting the number of evacuees within its sensing coverage. Based on the number of evacuees in each sensor s_i's neighborhood, the sensor maintains a potential value to find a navigation direction toward its neighbors. A sensor with the larger potential value implies that there are more evacuees in its vicinity. Therefore, each sensor will select the neighbor sensor with the minimal potential value to be its navigation direction. The potential value of each sensor s_i is computed by $P(s_i) = D(s_i) + \alpha \times P_g(s_i) + \beta P_N(s_j) + \gamma$, where $D(s_i)$ is the number of evacuees detected by s_i, $P_g(s_i)$ is the potential value of s_i's current navigation direction, and $P_N(s_j)$ is total number of evacuees detected by s_i's neighbors. Here, α, β, and γ are system parameters. By considering the relationship between the evacuee density and the walking speed of evacuees, the work in [15] extends [14] to reduce congestion, where the evacuee density is acquired by technologies such as image processing or RFIDs. This work adopts a discrete mapping function from the evacuee density to walking speed, where the evacuees' walking velocity is determined by changes in evacuee density. Thus, each sensor s_i computes its potential $P(s_i) = \Sigma_{\forall s_e \in S_E}(\frac{V_i}{d(s_i, s_e)} \times \frac{W_e}{\sum_{\forall s_e} W_e})$, where S_E is the subset of sensors at exits, $d(s_i, s_e)$ is the Euclidean distance between s_i and s_e, and W_e is a constant representing the exit capacity at exit s_e which depends for instance on the width of the exit. A larger potential value implies a path with the less congestion. Thus, each sensor will select the neighbor with the maximal potential value to be its navigation direction.

2.2. Geometric approaches

This type of approach exploits the unique properties of geometric graphs to plan evacuation paths as far from hazards as possible. For instance in [16] *Delaunay triangulations* [17] are used to partition a WSN into several triangular areas for planning area-to-area navigation paths, as shown in Fig. 3; each sensor knows its location and will cooperate to compute a planar graph, termed the *localized Delaunay triangulation* [18], in a distributed manner. For each triangulation, each sensor

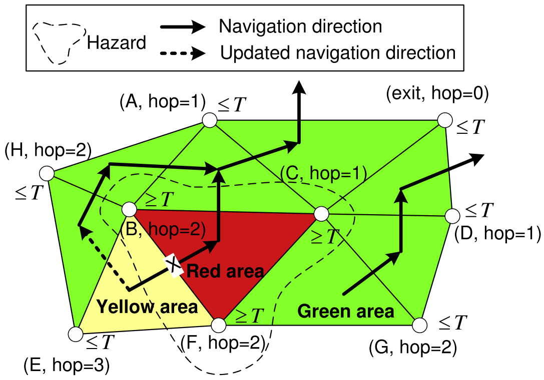

Fig. 3. An example of area-to-area navigation paths. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

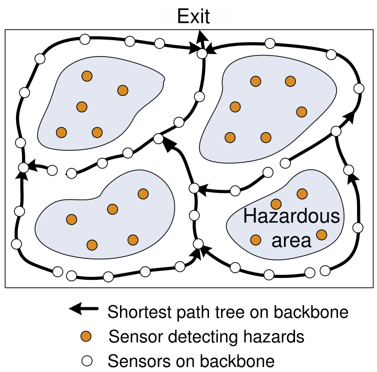

Fig. 4. An example of the backbone of road map.

maintains the following information: (i) another two sensors involved in the triangulation, (ii) one outgoing edge, and (iii) incoming edges. The initial outgoing edge of a triangulation is composed of the two sensors which are crossed by the navigation direction toward the neighboring triangulation with the smallest number of hops to the exit. For example in Fig. 3, $\triangle CFG$'s outgoing edge is $\overline{CG}$ since D has a smaller hop count to the exit than B. Triangulations will be of three colors based on sensor readings to represent different degrees of hazard. A triangulation is red if all of the three sensors of the triangulation detect a hazard (i.e., sensing reading is not smaller than a predefined threshold T). It is a yellow area if the two sensors on the outgoing edge detect a hazard but the remaining one does not. Otherwise, a triangulation is green. The navigation direction of each triangulation is then recomputed in both yellow and red areas to guarantee that the evacuees are guided toward green areas.

However, location information of both sensors and users may not always be available, so [19] maintains a road map in each user device to compute navigation paths, where the road map is a simple graph that represents the geometrical features of the environment. By measuring the signal strength, each evacuee follows a sequence of sensors to the exit along the safest navigation path. Based on the distance of each sensor from the hazardous areas, the backbone of the road map composed of a set of sensors, termed the *medial axis* [20], is created as shown in Fig. 4. Then, a shortest path tree of those sensors on the backbone is constructed; it is rooted at the exit so that evacuees do not go through hazardous areas. The navigation principle is to guide each evacuee to the sensors on the backbone and then to the exit along sensors on the backbone.

2.3. Prediction-based approaches

While most of the research focuses on finding paths based on real-time information, this type of approach uses the prediction of how long a hazard will take to reach the sensor, to compute the evacuation path with the longest escape time before the hazard reaches it. The work [21] predicts safe evacuation paths by maintaining two graphs, the *hazard graph* and the *navigation graph*. For each room and each line-of-sight corridor one sensor node is deployed. Each sensor has a corresponding vertex in both of the hazard graph and the navigation graph. Since hazards may spread across walls or corridors, there is an edge in the hazard graph between any two adjacent room-to-room, room-to-corridor, and corridor-to-corridor sensors. Also, there is an edge between two sensors in the navigation graph when there is a physical walking path between the two sensors. In the hazard graph, each edge between sensor s_i and sensor s_j will be associated with a hazard weight which is the estimated time for the hazard to spread from s_i to s_j. In the navigation graph, each edge between sensor s_i and sensor s_j will be associated with a navigation weight which is the estimated time for moving from s_i to s_j. By performing a breadth first search from the hazard source, each sensor can estimate a hazard time for itself.

The estimated safety of a navigation path P from a sensor s_i to the exit is the minimum of the difference between the hazard time of each node in P and the moving time from s_i to each sensor along P. Thus, each sensor can select the safest path toward the exit. The hazard time and also the moving delay time are considered in [22]. The delay time of each sensor is the estimated time for moving from the sensor to the exit that depends on the distance between the sensor node to the exit. According to the predefined hazard time threshold H_t and delay time threshold D_t, each sensor switches between five states: (1) $0 <$ hazard time $\leq H_t$ and delay time $\geq D_t$, (2) hazard time $> H_t$ and delay time $\geq D_t$, (3) hazard time > 0 and delay time $\leq D_t$, (4) delay time $= 0$, and (5) hazard time ≤ 0, each of which has different degrees of safety and guiding policies to each other. The guiding policy for sensors with state 1 is to send the evacuee further from the hazard and then head towards the exit. Sensors with state 2 are far from the hazard and the exit, so the guiding policy is based on the distance to the exit and to the fire. To reduce the density of evacuees near the exit, the guiding policy of state 3 is to navigate evacuees straight toward the exit. Finally, the state 4 refers to a node at the exit, while state 5 implies that the sensor is in a hazardous location. To reduce computation and communication overhead, each state has a different frequency of information update.

The inherent limitations of simple homogeneous WSNs as the sole means of supporting the communication and guidance needs of emergency evacuations has major shortcomings, leading to the need for *heterogenous cyber–physical emergency evacuation systems* to achieve essential requirements of real-time monitoring and quick response to emergencies. Such systems carry out monitoring and decision-making by considering two subsystems, a sensing subsystem and a decision support subsystem to overcome the inherent limitations of a WSN.

1. *Inherent limitations of sensors*: Complicated tasks may be too costly and beyond the sensors' capabilities of weak communication, storage, and computing. For example, an IEEE 802.15.4-compliant MICAz [23] has a 128 kB program flash memory, 512 kB measurement flash, 4 kB EEPROM, and 250 kbps data rate. However, the response time needs of an emergency system is much more strict than other WSN-aided applications. WSN-based emergency response systems with both sensing and computation tasks running on lightweight sensor nodes may incur unnecessary service delay. In contrast, separating sensing and decision-making tasks is more likely to achieve the requirement of quick response.
2. *Energy-efficient data collection*: Sensing capability is a necessary condition of emergency response systems. Since communication is energy-hungry and will consume more energy than computation especially for multihop relaying, reducing sensor energy consumption for relaying data [24,25] will prolong lifetime and is important. Thus, to prolong system lifetime in emergencies, separating the sensing and computational tasks is necessary for sensors so as to monitor physical information by single-hop collection with lower transmission power for energy-conservation purpose.
3. *Participatory sensing and opportunistic communication*: WSN coverage and connectivity may not be always achievable in dynamic and uncertain emergencies. User personal devices may be used to overcome these limitations, where available sensors and users devices would cooperate to collect and disseminate physical information such as hazard spreading and the location of people, and make distributed decisions.
4. *Faster-than-real-time prediction and forecast*: Prediction and forecast can avoid unnecessary casualties but are difficult for lightweight sensors. A separated decision support subsystem can serve as a predictor to compensate for the limitations of the sensing subsystem so as to forecast dynamic changes in emergency.
5. *Multi-dimension responses*: while in-situ and real-time information is collected by different types of sensors, selecting the best action and response based on modeling and optimization will be needed.

2.4. Decentralized evacuation systems

This type of system is a two-tiered architecture composed of a distributed decision system and a sensing system. The upper-tier distributed decision system serves as a middleware layer to connect two different types of client in the lower tier, the sensing clients and the user clients. Note that all sensor nodes do not form a connected WSN and are only responsible for reporting in-situ and real-time information to the decision system. In the work reported by [26], there are two major components, the *sensing component* and the *decision component*. The sensing component is composed of a set of sensor nodes, while the decision component is composed of a set of lightweight decision nodes (DNs). Each sensor node is responsible for reporting hazard intensity to the neighboring decision nodes. These DNs are deployed in predetermined locations in the building (e.g., rooms and corridors) to form a distributed network so as to compute the safest evacuation paths. The ideas here are inspired from adaptive QoS-aware algorithms for packet routing that were described earlier in [27]. The idea is to replace the "packet" in a packet network by an "evacuee", the QoS in the packet network is replaced by a measure combining delay to the safe exit and the safety of the path, and the DNs play the role of routers for the evacuees. For a recent survey of experimental results concerning QoS-aware routing algorithms in packet networks, see [28]. To make this scheme workable in the evacuation scenario, each evacuee is equipped with a user portable device (e.g., a smart phone) which communicates with the DNs directions. The potential paths that evacuees can follow in the physical system mimic the paths in the network, and the DNs are physically placed at decision points of an undirected graph which represents the displacements that evacuees can make as they move in the building. Edges between DNs represent short physical distances that evacuees can move through, and the distances are short enough so that adjoining DNs can communicate very reliably with each other. The decision algorithm will operate in a distributed fashion at all of the DNs, and they will all have stored this simple graph representation. In the algorithm, a cost or effective length associated with the edge will relate to the physical distance, and will be increased as a function of hazard based on sensed information, that will be propagated progressively among DNs. The *effective length*, which represents the degree of hazard of this particular edge based on the physical length and sensing information. is be computed as $d(D_i, D_j) \cdot H(D_i, D_j)$, where $d(D_i, D_j)$ is the physical distance between DNs D_i and D_j and $H(D_i, D_j)$ is the hazard intensity reported by the sensor placed on this edge. Using adaptive routing techniques [29], each DN can find the neighbor DN along the minimum-cost path to an exit. By considering the *spatial correlation* of hazards, in the sense that the neighborhood of a hazard may be more dangerous than the other areas further away from the hazard, the work in [30] extends [26] to a more realistic emergency scenario, where the hazard intensity reported by each sensor aggregates its own value with that of neighboring sensors. Thus the effective length between DN D_i and DN D_j is computed as $d(D_i, D_j) \cdot H_s(D_i, D_j)$, where the $H_s(D_i, D_j) = H(D_i, D_j) + H_{\text{avg}}$. Here, $H(D_i, D_j)$ is the hazard intensity monitor by the sensor at the edge (D_i, D_j), denoted by s_{ij}, and H_{avg} is the average hazard intensity monitored by s_{ij}'s neighbors.

However DNs may fail or be destroyed during an emergency, so that in [31] *opportunistic communications* are suggested for the design an elastic evacuation system, where a set of mobile decision nodes (MDN) that are carried by the evacuees and possibly by emergency personnel are used to complement or partially replace the DNs.

Each MDN maintains a navigation graph as explained in [26]. Static sensor nodes are pre-deployed in the building to perform localization of MDNs and report their sensing information to those MDNs that are passing within communication range of sensors. These MDNs will form an opportunistic network in the sense that the information is acquired and disseminated only when MDNs can contact each other. When two MDNs meet within their communication range, a low latency flooding-based information dissemination policy [32]. Once a MDN acquires the newest information from static sensor nodes or other MDNs, it will perform a local update on link costs in the navigation graph to recompute the safest path for the evacuee. The work of [33] considers a *hybrid decision component*, where both static decision nodes and mobile decision nodes may coexist for increased reliability and message delivery rate.

In [34], the concept of functional separation is adapted to partitioning the sensing system into two sensing units, the *emergency sensing unit* and the *position sensing unit*. The emergency sensing unit is composed of different types of sensor node (e.g., thermometer, hygrometer, vision sensors, and microphones) to sense abnormal events, while the position sensing unit consists of users' mobile phones to sense the wireless beacons for localization. Each mobile phone fetches sensing data from a centralized *sensor-data management middleware* that is a sensing information pool to conduct localization and evacuation navigation. Apparently, the functionality-separated concept is also considered as a means to isolate a sensing database from the sensor clients and user clients.

3. Agent-based simulators

Agent-based emergency response simulators [35,36] have attracted much attention on facilitating decision of first-aid responders in an emergency, where the physical world is modeled as fixed-distance grid points. To reduce the computation cost, [37] models the physical word as a directed graph $G = (V, E)$. The vertex set V is composed of a set of *Points of Interest (PoI)*, each of which is corresponding to a location in the physical world (e.g., a room, a segment of a corridor, a door, and a stair). There is an edge in E between two PoIs if there is a motion path between the two locations in the physical world. In the simulator, each PoI maintains a few attributes including ID, the 3D coordinate, the PoI type (e.g., room or corridor), the availability (e.g., wireless connection). Similarly, each edge maintains attributes including ID, the two end points of PoIs, the length, the type, the degree of risk. Each dynamic entity (e.g., evacuee, robot, or rescuer) is regarded an agent that participates in the simulated event. To simulate movements of agents, each agent has a moving function of PoIs which considers three factors, the duty, the risk, and the imitation. The factor of duty is an evaluation of the attractive force to a PoI (e.g., the exit has attractive force on each evacuee) so that a dutiful agent will generally be more responsive to instructions it receives. The factor of risk evaluates the perceived danger of the agent moving toward a PoI which will depend on the condition of the path. Imitation is the evaluation of influence degree among agents which is to avoid consistency among agents. By considering that each entity in the physical world only has partial knowledge, each agent updates the attribute values only when it arrives to a PoI. Note that more realistic physical sensor inputs are adopted in this simulator to simulate hazards so that the simulator can also provide real-time decisions for events that take place in the physical world.

To speed up the simulations in a complicated environment, DEFACTO [38], SimSITE [39], and DBES [40] consider a distributed emergency simulator. The goal of DEFACTO [38] is to provide a 3D visualization system for first-aid decision makers to facilitate emergency response in a large-scale outdoor environment. There are two components in DEFACTO: the *Omni-Viewer* and the *proxy-based teamwork*. The Omni-Viewer is a 3D human interface to facilitate interactions between first-aid experts and the environment, while the proxy-based teamwork consists of distributed proxies, each of which controls and coordinates a partial of agents in the simulated environment. DEFACTO does not follow the specification of the High-Level Architecture (HLA) [41] which specifies a general framework to address scalability issues on distributed simulations. As following HLA, SimSITE [39] also built a 3D visualization simulator to facilitate training of first-aid responders, where the physical world is modeled as a matrix so as to compute the shortest path for each evacuee along a set of consecutive entries in the matrix. Instead of the grid-point model, graph-based DBES [40] follows HLA to extend [37] to a distributed evacuation simulator. In a distributed simulator, how to decompose the simulated environment into several parts without incurring too much communication and computation cost is a key issue. Since the simulated events (e.g., hazard spreading) may have temporal and spatial independence in an emergency evacuation application, an application-driven decomposition manner is adopted in this work, where a multi-stair building could be partitioned into multiple

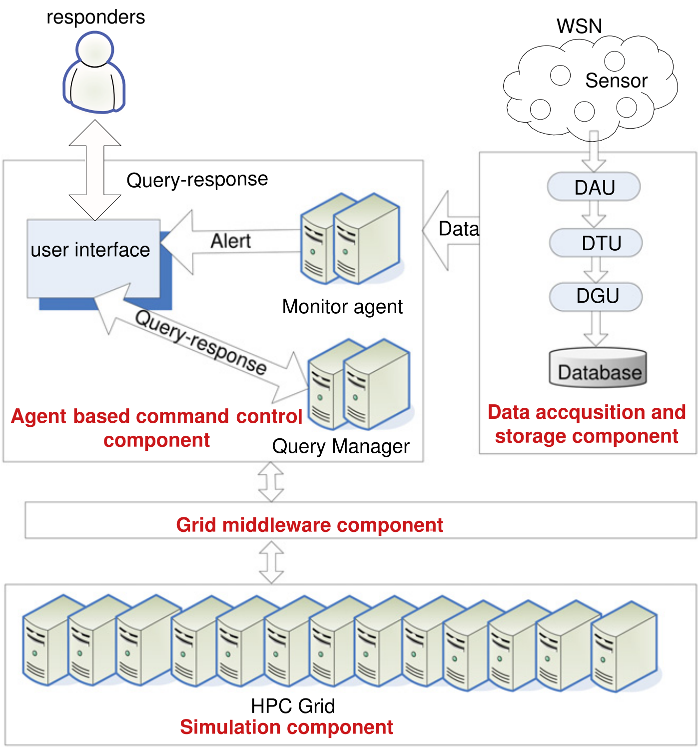

Fig. 5. The FireGrid architecture.

floors or groups of rooms. Two types of agents, *simulators* and *simulated agents*, are designed in DBES. The simulators are a set of distributed event schedulers, each of which is associated with a smaller part in the environment to coordinate the interactions among events. The simulated agents are the set of active units, each of which is associated with an entity in the physical world and be implemented a specific goal and strategy. To reduce storage, each individual simulator only maintains a brief of other simulators in a condensed way (e.g., a set of PoIs at the intersection of corridors or a stairway). To reduce internal communication costs within the distributed simulator, a simulated agent's attributes will migrate from one simulator to another only when the simulated agent migrates between the physical areas that are being represented by each part of the simulator. To facilitate the communication among all agents, the FIPA handshaking protocol [42] among agents is adopted.

Some papers have considered the social interactions and behavior among evacuees to design simulators. For instance in [43] a virtual–physical emergency simulator for a city-wide environment introduces the concept of *participatory simulations* in the sense that all evacuees are involved in the process of the simulation to interact with each other. This system is composed of two major agents, the evacuee agent and the guide agent. The evacuee agent is responsible for simulating evacuees in the virtual city map, while the guide agent is responsible for providing navigation instructions for evacuees. Instead of having homogeneous characteristics for all evacuees, in [44] each individual evacuee has unique characteristics in their physical, psychological, and motion aspects. In this simulator, each evacuee is an agent with physical attributes, psychological attributes and moving attributes. Based on these characteristics, Particle Swarm Optimization (PSO) [45], is adopted to make decisions regarding the evacuation.

3.1. *Grid-supported systems*

This type of system relies on the technology of *Grid Computing* to assimilate multiple types of sensing data so as to provide real-time and faster-than-real-time prediction for emergency responders, where the functionality-separated sensing system, storage system, and computation collaborate to support emergency response. Such systems will often rely on simulation tools in order to provide fast prediction.

The *FireGrid* system is described in [46]. It consists of four major components: (1) the *data acquisition and storage*, (2) the *simulation component*, (3) the *agent-based command-control component*, and (4) the *Grid middleware*, as shown in Fig. 5. The data acquisition and storage component is responsible for collecting and storing sensing information through the three functional units as follows. The Data Acquisition Unit (DAU) collects raw sensing data from multiple types of sensors (e.g., smoke, temperature, and gas sensors). The Data Translation Unit (DTU) transforms the raw sensing data into an adequate form (e.g., transforming thermocouple voltage readings into temperatures). The Data Grading Unit (DGU) filters the information to make sure about the accuracy and reliability of the database. The simulation component provides a parallel and distributed computing resource to interpret the current status and predict the future behavior of an ongoing fire.

4. Search and rescue systems

Identifying the locations of the possible victims is usually the first step before rescue. An overestimate of victims may lead to resource wastage and more serious casualties. Ref. [47] studies a double-counting problem in a WSN due to overlaps of sensing coverage. Given a set of camera-based sensors, each of which is capable of counting how many targets (i.e., victims) are in its sensing coverage, this work aims at finding the probability mass function of the total number of targets. By exhaustively enumerating all of combinations of non-overlapping areas partitioned by the sensing circles, the probability of a given number of targets could be found based on the conditional probability of all possible cases. However, exhaustive enumeration will incur extremely high complexity in computation which depends on the number of non-overlapping areas. Thus, this work proposes a low-complexity counting mechanism by reducing the enumerating space, where the zero-detection sensing area will be eliminated first and the remaining non-overlapping areas are divided into several groups with the even number of areas for counting independently. So, the final probability of a given total number of targets could be estimated by combining the probabilities of these groups. To reduce counting error, the division of groups is based on the size of the areas or the number of targets in areas.

Injured civilians who are immobilized will need to communicate with the external world when the communication infrastructure fails, and [48] addresses robot deployment to connect as many civilians possible with a static base station. Assume that each civilian is equipped with a mobile device with a communication range of r_c and each robot has a communication of R_c. For a given candidate positions of robots which may depend on the destroyed degree of the environment and robots' capabilities, the work formulates the robot deployment problem as a mixed integer linear programming such that the number of connected civilians is maximized, where each connected civilian can communicate with the static base station in a multi-hop way. Considering the uncertainty of civilians' locations, [49] extends [48] to a distributed heuristic, where the number of civilians at a given position u is modeled as a random variable X_u. Thus, given a probability threshold q, for each candidate position u, we can compute the expected number of civilians $E_q(u) = E_q(X_u|X_u < m)$. Based on the uncertain model, an iterative heuristic is designed in a greedy way. For each iteration, each robot only considers a subset of candidate positions with $E_q(u) > L$ and applies k-means clustering to group these candidate positions into k clusters with radius smaller than $r_c + R_c$ for the connectivity guarantee, where k is an adequate value. Each robot a_i moves toward the cluster C with the maximum attraction $\frac{\sum_{u\in C} E_q(u)}{d(a_i,C)}$ to compete for being cluster head, where $d(a_i, C)$ is the moving distance from a_i's current position to cluster C. A cluster-head robot a_i must issue an exploration message to request other robots to connect the civilians within its cluster. Once all civilians in a cluster have explored, the cluster-head robot reduces the threshold of the considered subset to avoid attracting other robots. In this way, robots can iteratively explore civilians as many as possible. While many efforts focus on the coverage and connectivity issues in a WSN [50], this type of work concentrates on discovering possible communication devices in a highly uncertain environment for search and rescue purposes. Comprehensive solutions to moving planning of robots are presented in [51].

Considering the changes of victims' locations in an uncertain environment (e.g., aftershocks sites may shift after a large earthquake), [52] proposes a robot-sensor network system for tracking victims autonomously without relying on localization technologies. Assume that robots have the capability of estimating the distance between itself and the neighboring sensors, and victims will generate detectable signals such as heat, CO_2, or sounds. Inspired by the thermotaxis of insects, the main idea is to make the movements of robots from *colder* sensors toward the *hotter* sensors (i.e., closer to victims). Each sensor will maintain a gradient to facilitate the moving of robots. If a sensor s_i detects a victim, it will set its gradient as $g_i = 0$, and broadcast it. While sensor s_i receives broadcast messages, it will set its gradient as $\max\{g_j|s_j \in N(s_i)\} + 1$, where $N(s_i)$ is the set of neighbors of s_i and also broadcast it. Each robot will switch between three states: (1) blind search, (2) follow sensors, and (3) approach the victim. A robot will switch from state 1 to state 2 (resp. to state 3) when it receives sensors' (resp. to victims') signals. A robot in state 2 will move toward the neighboring sensor with the lowest gradient until it receives a signal from a victim. A robot in state 3 must keep approaching the victim unless it loses the signal from the victim or other robots are tracking the victim. Next, we will discuss the optimization of resource allocation in an emergency, where limited resources (e.g., firefighters or ambulances) need be assigned to several events (e.g., fire events or injurious people) to conduct emergency response. Two types of approach in an environment with certainty and uncertainty will be reviewed. In an environment with certainty, each resource is assumed success in executing a mission once it has been allocated to an event. In contrast, in an uncertain environment, a resource may fail in executing a mission.

Ref. [53] exploits the Voronoi polygons to design a selection scheme of mobile resources (''*resource*'' for short). All resources will form Voronoi polygons, each of which contains exactly one resource, where each sensor in a particular Voronoi polygon is closer to the resource than to other resources. When a static sensor s_i needs to request for a resource, it will send a weight request packet (WREQ) to look for an adequate resource. Once receiving a WREQ, each resource a_j will reply its weight $w_j = \frac{\text{Area}_j \times h(s_i,a_j)}{e_j}$ to bid for the event, where the Area_j is the area of a_j's Voronoi polygon, $h(s_i, a_j)$ is the hop distance from s_i to a_j, and e_j is a_j's energy. A resource with the smaller weight will win the bidding and move to the event location for providing services.

Instead of geometric approaches, [54] considers a grid-quorum network to design a service discovery scheme. Each service provider (i.e., resource) disseminates its location information along a ''column'' in the network. An event will search resources along a ''row'' in the network. In this way, service delivery is guaranteed since there is at least one intersection of column-based dissemination and row-based searching. However, the computation of Voronoi polygons must rely on the

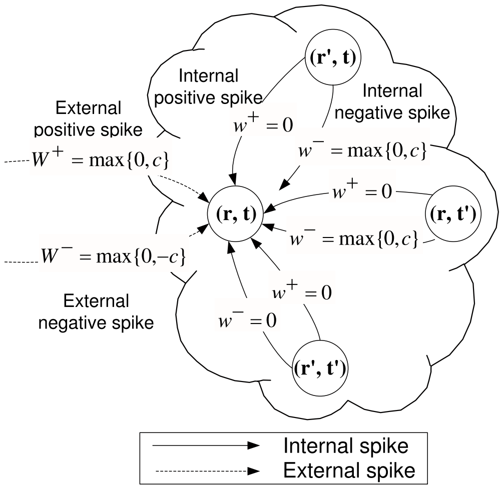

Fig. 6. The model of RNN for the resource allocation problem.

location information of nodes, and the quorum-based approaches may incur network-wide flooding in case all resources are collinear. Ref. [55] combines the concepts of Voronoi polygons and the quorum-based approaches to design a localized service discovery scheme. Assuming that all nodes form a grid-topology network, each resource disseminates its location information along its residing row and column but broadcast messages will be blocked at the nodes which have received a message from another resource with smaller hop distance. In this way, the network will be partitioned into several cells. Each event will request a service from the nearest resource by disseminating discovery messages along its residing row and column within its cell.

By considering the energy issue, in [56] resources are dispatched to events in an energy-balanced way. Each event will search resources within a limited radius to find the one with the minimum energy consumption on moving. If no resource is found, the searching radius will be incrementally increased until each event has be matched with a resource. Considering that people may be trapped with no escape path, [57] proposes a mechanism to dispatch resources to eliminate key hazards or obstacles first so that the number of rescued victims is maximized. By modeling a WSN as a network flow problem [58], this work finds the minimum cut, which is a subset of edges with the smallest total capacity than other cuts, to allocate resources for the elimination of hazards or obstacles.

The complexity of resource allocation in emergency situations requires fast and near-optimal algorithms to provide advice in real-time. Thus in [59] the related resource allocation problems are formulated as a non-linear optimization problem, which is then solved approximately in real-time using a previously trained Random Neural Network [60,61]. Note that the generic resource allocation problem is NP-hard for both deterministic and nondeterministic cases, so this work proposes polynomial-time approximate solutions. Considering the generic ambulance allocation problem, a set of tasks (denoted by T) represent a set of emergencies to which ambulances must be sent, while a set of resources (denoted by R) are the ambulances themselves. The initial locations of both tasks and resources are known and the traversal cost (distance or time) can be estimated by each resource. A task is allowed to be executed by multiple resources, while a resource cannot be assigned to multiple tasks at the same time. The optimization problem must minimize the total execution cost $C_{\text{total}} = \sum_{t\in T}\sum_{r\in R} C(r,t)p(r,t) + \sum_{t\in T} K(t)\prod_{r\in R}(1-(1-q(r,t))p(r,t))$. Here, $p(r,t)$ and $q(r,t)$ are the probabilities of assigning resource r to task t and the probability that resource r will fail in executing task t respectively; $C(r,t)$ and $K(t)$ are the cost of resource r executing task t and the penalty of a failed execution for task t respectively. This optimization problem is to obtain the $p(r,t)$ for each $t \in T$ and $r \in R$. Note that for a deterministic case, $p(r,t) \in \{0,1\}$. To solve the problem, a RNN is constructed with $|R| \times |T|$ neurons, each of which has an internal excitation probability $p(r,t)$ which is a result of the RNN internal interconnections and external signals. Each neuron receives positive and negative spikes from other neurons with interconnect rates w^+ and w^- and from the external world with arrival rates W^+ and W^-, as shown in Fig. 6, where $c = K(t)(1-q(r,t)) - C(r,t)$ is the expected reduction of cost when r is allocated to t. For each neuron (r,t), the negative spike from (r',t) has a non-negative arrival rate to inhibit distinct resource r and r' being allocated to the same task t, and the negative spike from (r,t') also has a non-negative arrival rate to inhibit resource r being allocated to two distinct task t and t'. After the RNN is constructed, the unique solution to $p(r,t)$ for each $r \in R$ and $t \in T$ is obtained. Finally, based on the value of $p(r,t)$, a greedy heuristic is iteratively conducted to choose the largest positive $p(r,t)$ until there is no available resource.

4.1. Toward cyber–physical search and rescue platforms and systems

The Pathfinder [62] system provides firefighters with safety navigation during rescue missions and two major components, ultrasonic beacons and ultrasonic trackers, are adopted. Each firefighter wears an ultrasonic tracker to receive signals from ultrasonic beacons with three types of ultrasonic beacons: the firefighter beacon, the exit beacon, and the auxiliary beacon, each of which works on a different frequency for different purposes. Each firefighter wears firefighter beacons so that injured firefighters can be found by other firefighters. Exit beacons are used to mark exits, while auxiliary beacons are used to mark way-points inside a building or injured/trapped people along a return path. In [63] the *CenWits (Connection-less Sensor-Based Tracking System Using Witnesses)* is proposed to search for lost or injured hikers in a large wilderness area. Instead of a well-connected network, all hikers form an opportunistic network to exchange their *witness* information to indicate encounters with each other so as to provide approximate areas of disappearance of hikers for rescuers. The system consists of a number of sensors, access points (APs), location points (LPs), and an external processing center. Each hiker carries a sensor with a GPS receiver and an RF transmitter for communicating with other sensors, APs, and LPs. A set of APs are deployed at predefined locations where the hikers may pass through (e.g., intersections of footpaths or resting areas), connected to the external processing center. A few LPs are deployed at particular locations to update sensor locations in case the GPS cannot work. The external processing center is responsible for collecting the witnesses from all APs. When two sensors meet in their communication range, they will record the presence of each other in their witnesses and also exchange their earlier witnesses, where each record in the witnesses including encountered node ID, the current time, encountered location, the number of transferred hops. Once a sensor meets an AP, all of its witnesses will be uploaded to the AP. Based on the witnesses, the system can estimate the possible lost locations of a hiker to perform rescue missions. This work points out two technical challenges, memory and power management for maintaining witnesses at each sensor. For memory management, a record will be deleted at a sensor based on the number of transferred hops or the record gap. For power management, hikers could be separated into groups, each of which has a group leader, based on social or geographical relationships, and witnesses are only exchanged among group leaders.

Ref. [64] describes a navigation system, termed *robot-and-sensor team*, to control moving robots along safe paths. This system consists of static temperature sensors for monitoring, static radio tags for localization, and mobile robots. Based on the collected sensing data, a temperature gradient graph is constructed to navigate robots' movements. Three technical challenges, localization, information flow of navigation, and asynchronous network control, are studied. SmokeNet [65] tracks firefighters and provides safe navigation paths for firefighters in a multi-stair building. The system consists of a static WSN with a sink, static beacons, wearable beacon receivers, head-mounted displays, and an incident command center. The WSN serves as a communication backbone to connect the outside incident command center. Each wearable beacon receiver is responsible for the localization of the firefighter in a RF-fingerprint technology. The head-mounted display is a user interface which can see floor map and locations of other firefighters. The incident command center provides key information (e.g., firefighters' location, hazard status, etc.) to outsider hummer commanders. To guide people, sensors are integrated with red, yellow, and green LEDs to serve as traffic lights so as to indicate safe paths (green lights), dangerous paths (red lights), and the system out of service (yellow lights). This work conducts comprehensive experiments to investigate localization accuracy communication reliability in highly dynamic and uncertain emergencies.

Ref. [66] focuses on designing hardware platforms of robots to facilitate monitoring, searching, and communication between rescuers and victims. This work designs a controllable guiding robot with two DC motors, a RC servomotor, a camera, multiple types of sensor (e.g., temperature, CO, O_2, gas, and compass), a speaker, a microphone, an LED-set lamp, and an RF module. These controllable robots could be thrown into narrow space to serve as searchers in an emergency. In case victims are discovered by robots' cameras, the rescuers can communicate with victims via the robots. Three major challenges of designing robots are raised: small size, temperature protection, and waterproofing.

The work [67] implements a robot tracking system in an outdoor emergency (e.g., earthquakes), where robots serve as searchers to discover victims and a static WSN facilitates localization and tracking of the robots. Each robot is equipped with a ZigBee communication module for reporting sensing data and localization. Each static sensor serves as a reference node for localization, while each robot will query neighboring sensor nodes to measure received signal strength and estimate the location of itself. To improve localization accuracy, a motor encoder and an electronic compass are installed in the each robot, where the motor encoder is to measure the displacement of the robot based on the number of the motor's rotations and the electronic compass is to obtain the direction of the robot.

LifeNet [68] provides an electronic lifeline for firefighters to mark the paths they have taken to facilitate the process of search and rescue. Each firefighter is equipped with a beacon ejector, a boot-mounted beacon receiver, a head-mounted display, and a wearable computer. The beacon ejector will deploy beacon nodes automatically along the firefighters' trails with a fixed time interval. The boot-mounted beacon receiver is integrated with a sensor node to collect beacon signals and environmental information, where the localization relies on the ultrasonic ranging. The head-mounted display provides two modes for firefighters. The first mode is to show the direction of a retreat path marked by beacon nodes, while the second mode is to show the detected beacon nodes and other firefighters within its range. A challenge of the system is the issue of beacon movement since a fixed-distance deployment of beacon nodes is provided by the beacon ejector.

4.2. Summary of search and rescue systems

Refs. [52,67] focuses on searching for victims or tracking their movements, while [62,65,68] pays attention to tracking rescuers (or firefighters) to guarantee their safety. Based on the presence of victims, [63] focuses on reducing search and rescue space. To connect with civilians in an emergency, [48,49] form a communication backbone, while [64,66] uses robots to search for and guide victims. Ref. [47] assesses the number of civilians in an emergency, while [53–57,59] concentrate on the optimization of search and rescue cost.

5. Beyond existing technologies

Beyond existing technologies, this paper has compared the various trends of research on emergency response systems by discussing the relationship between the level of technical support and how well an emergency response system can work to achieve their primary objective of saving lives and improving human wellbeing. One important research area in emergency management which has been neglected so far is the use of probability modeling, which has long been used in the study of large scale systems [69,70], as well as in distributed systems [71], and for evaluating and handling uncertainty in sources of information [72]. An advantage of such approaches is that they can provide a computationally fast mathematical probabilistic prediction of the overall performance of algorithms and policies, prior to lengthy simulation studies or experimental evaluations, even though the mathematical analysis may not be able to include all relevant aspects.

We think that such methods will become more important in this field, as a way to conduct predictions of the spread of hazards during an emergency with moving entities such as robotic resources and human beings. Such models are also useful in obtaining best case and worst case bounds, as well as the average behavior which is less useful in such situations. Probability models can also be used to evaluate risk and damage, and to model the effect of failures in the different components and infrastructures of the system being considered.

References

[1] Terrestrial ecology observing systems, center for embedded networked sensing. http://research.cens.ucla.edu/.
[2] F. Hu, Y. Xiao, Q. Hao, Congestion-aware, loss-resilient bio-monitoring sensor networking for mobile health applications, IEEE Journal on Selected Areas in Communications 27 (4) (2009) 450–465.
[3] H. Liu, P. Wan, X. Jia, Maximal lifetime scheduling for sensor surveillance systems with k sensors to one target, IEEE Transactions on Parallel and Distributed Systems 17 (12) (2006) 1526–1536.
[4] M. Tubaishat, P. Zhuang, Q. Qi, Y. Shang, Wireless sensor networks in intelligent transportation systems, Wireless Communications and Mobile Computing 9 (3) (2009) 287–302.
[5] P.-Y. Chen, W.-T. Chen, Y.-C. Tseng, C.-F. Huang, Providing group tour guide by rfids and wireless sensor networks, IEEE Transactions on Wireless Communications 8 (2) (2009) 3059–3067.
[6] D.J. Malan, T.R. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, S. Moulton, Sensor networks for emergency response: challenges and opportunities, IEEE Pervasive Computing 3 (4) (2004) 16–23.
[7] C. Fischer, H. Gellersen, Location and navigation support for emergency responders: A survey, IEEE Pervasive Computing 9 (1) (2009) 38–47.
[8] Q. Li, M.D. Rosa, D. Rus, Distributed algorithms for guiding navigation across a sensor network, in: ACM Int'l Conf. Mobile Computing and Networking, 2003, pp. 313–325.
[9] E. Gelenbe, K. Hussain, V. Kaptan, Simulating autonomous agents in augmented reality, Journal of Systems and Software 74 (3) (2005) 255–268.
[10] V. Kaptan, E. Gelenbe, Fusing terrain and goals: agent control in urban environments, in: Multisource Information Fusion: Architectures, Algorithms, and Applications 2006, vol. 6242, SPIE, 2006, p. 7179.
[11] Y.-C. Tseng, M.-S. Pan, Y.-Y. Tsai, Wireless sensor networks for emergency navigation, IEEE Computer 39 (7) (2006) 55–62.
[12] M.-S. Pan, C.-H. Tsai, Y.-C. Tseng, Emergency guiding and monitoring applications in indoor 3D environments by wireless sensor networks, International Journal of Sensor Networks 1 (1–2) (2006) 2–10.
[13] D. Chen, C.K. Mohan, K.G. Mehrotra, P.K. Varshney, Distributed in-network path planning for sensor network navigation in dynamic hazardous environments, Wireless Communications and Mobile Computing (2010).
[14] W.-T. Chen, P.-Y. Chen, C.-H. Wu, C.-F. Huang, A load-balanced guiding navigation protocol in wireless sensor networks, in: IEEE Global Telecomm. Conf., 2008, pp. 1–6.
[15] P.-Y. Chen, Z.-F. Kao, W.-T. Chen, C.-H. Lin, A distributed flow-based guiding navigation protocol in wireless sensor networks, in: Int'l Conf. Parallel Processing, September 2011, pp. 105–114.
[16] P.-Y. Chen, W.-T. Chen, Y.-T. Shen, A distributed area-based guiding navigation protocol for wireless sensor networks, in: IEEE Int'l Conf. Parallel and Distributed Systems, 2008, pp. 647–654.
[17] F.P. Preparata, M.I. Shamos, Computational Geometry: An Introduction, Springer-Verlag, 1985.
[18] X.-Y. Li, G. Calinescu, P.-J. Wan, Y. Wang, Localized Delaunay triangulation with application in ad hoc wireless networks, IEEE Transactions on Parallel and Distributed Systems 14 (10) (2003) 1035–1047.
[19] M. Li, Y. Liu, J. Wang, Z. Yang, Sensor network navigation without locations, in. IEEE INFOCOM, 2009, pp. 2419–2427.
[20] J. Bruck, J. Gao, A.A. Jiang, MAP: medial axis based geometric routing in sensor networks, in: ACM Int'l Conf. Mobile Computing and Networking, 2005, pp. 88–102.
[21] M. Barnes, H. Leather, D.K. Arvind, Emergency evacuation using wireless sensor networks, in: IEEE Conf. Local Computer Networks, 2007, pp. 851–857.
[22] A. Cherniak, V. Zadorozhny, Towards adaptive sensor data management for distributed fire evacuation infrastructure, in: Int'l Conf. Mobile Data Management, 2010, pp. 151–156.
[23] Micaz. http://www.memsic.com/.
[24] M. Zhao, M. Ma, Y. Yang, Mobile data gathering with space-division multiple access in wireless sensor networks, in: IEEE INFOCOM, 2008, pp. 1283–1291.
[25] G. Xing, T. Wang, Z. Xie, W. Jia, Rendezvous planning in wireless sensor networks with mobile elements, IEEE Transactions on Mobile Computing 7 (12) (2008) 1430–1443.
[26] A. Filippoupolitis, E. Gelenbe, A distributed decision support system for building evacuation, in: IEEE Int'l Conf. Human System Interactions, 2009, pp. 320–327.
[27] E. Gelenbe, Z. Xu, E. Seref, Cognitive packet networks, in: ICTAI, 1999, pp. 47–54.
[28] E. Gelenbe, Steps towards self-aware networks, Communications of the ACM 52 (2) (2009) 66–75.
[29] E. Gelenbe, Cognitive packet network (CPN). US, Patent 6,804,20, US, Patent Patent 6,804,201, October 11, 2004.
[30] A. Filippoupolitis, E. Gelenbe, An emergency response system for intelligent buildings, in: Sustainability in Energy and Buildings, Springer, 2011, pp. 265–274.
[31] G. Gorbil, E. Gelenbe, Opportunistic communications for emergency support systems, in: Int'l Conf. Ambient Systems, Networks and Technologies, 2011, pp. 1–9.
[32] A. Vahdat, D. Becker, Epidemic routing for partially-connected ad hoc networks, Duke University Department of Computer Science, Tech. Rep., 2000.
[33] G. Gorbil, A. Filippoupolitis, E. Gelenbe, Intelligent navigation systems for building evacuation, in: Computer and Information Sciences, Lecture Notes in Electrical Engineering, Springer, 2011, pp. 339–345.
[34] Y. Inoue, A. Sashima, T. Ikeda, K. Kurumatani, Indoor emergency evacuation service on autonomous navigation system using mobile phone, in: Int'l Symp. Universal Communication, 2008, pp. 79–85.

[35] V. Balasubramanian, D. Massaguer, S. Mehrotra, N. Venkatasubramanian, DrillSim: a simulation framework for emergency response drills, Intelligence and Security Informatics 3975 (2006) 237–248.
[36] Robocuprescue. http://www.rescuesystem.org/robocuprescue/.
[37] A. Filippoupolitis, L. Hey, G. Loukas, E. Gelenbe, S. Timotheou, Emergency response simulation using wireless sensor networks, in: Int'l Conf. Ambient Media and Systems, 2008, pp. 21:1–21:7.
[38] N. Schurr, J. Marecki, M. Tambe, P. Scerri, N. Kasinadhuni, J. Lewis, The future of disaster response: Humans working with multiagent teams using DEFACTO, in: AAAI Spring Symp. AI Technologies for Homeland Security, 2005.
[39] K. Liu, X. Shen, A.E. Saddik, A. Boukerche, N.D. Georganas, SimSITE: The HLA/RTI based emergency preparedness and response training simulation, in: IEEE Symp. Distributed Simulation and Real-Time Applications, 2007, pp. 59–63.
[40] N. Dimakis, A. Filippoupolitis, E. Gelenbe, Distributed building evacuation simulator for smart emergency management, The Computer Journal 53 (9) (2010) 1384–1400.
[41] IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)-framework and Rules, 2000.
[42] The foundation for intelligent physical agents (fipa). http://www.fipa.org.
[43] Y. Nakajima, H. Shiina, S. Yamane, T. Ishida, Disaster evacuation guide: Using a massively multiagent server and GPS mobile phones, in: IEEE Int'l Symp. Applications and the Internet, 2007.
[44] Y. Bo, W. Yong-gang, W. Cheng, A multi-agent and GIS based simulation for emergency evacuation in park and public square, in: Int'l Conf. Computational Intelligence and Security, 2009, pp. 211–216.
[45] J. Kennedy, R. Eberhart, Particle swarm optimization, in: IEEE Int'l Conf. Neural Networks, 1995, pp. 1942–1948.
[46] L. Han, S. Potter, G. Beckett, G. Pringle, S. Welch, S.-H. Koo, G. Wickler, A. Usmani, J.L. Torero, A. Tate, FireGrid: an e-infrastructure for next-generation emergency response support, Journal of Parallel and Distributed Computing 70 (11) (2010) 1128–1141.
[47] S. Guo, T. He, M.F. Mokbel, J.A. Stankovict, T.F. Abdelzaher, On accurate and efficient statistical counting in sensor-based surveillance systems, in: IEEE Int'l Conf. Mobile Ad Hoc and Sensor Systems, 2008, pp. 24–35.
[48] G. Loukas, S. Timotheou, E. Gelenbe, Robotic wireless network connection of civilians for emergency response operations, in: Int'l Symp. Computer and Information Sciences, 2008, pp. 1–6.
[49] S. Timotheou, G. Loukas, Autonomous networked robots for the establishment of wireless communication in uncertain emergency response scenarios, in: ACM Symp. on Applied Computing, 2009, pp. 1171–1175.
[50] A. Ghosh, S.K. Das, Coverage and connectivity issues in wireless sensor networks: a survey, Pervasive and Mobile Computing 4 (3) (2008) 303–334.
[51] J.-C. Latombe, Robot Motion Planning, Kluwer, New York, 1992.
[52] J. Reich, E. Sklar, Robot-sensor networks for search and rescue, in: IEEE Int'l Workshop Safety, Security and Rescue Robotics, 2006.
[53] A. Verma, H. Sawant, J. Tan, Selection and navigation of mobile sensor nodes using a sensor network, Pervasive and Mobile Computing 2 (1) (2006) 65–84.
[54] I. Stojmenovic, D. Liu, X. Jia, A scalable quorum based location service in ad hoc and sensor networks, International Journal of Communication Networks and Distributed Systems 1 (1) (2008) 71–94.
[55] X. Li, N. Santoro, I. Stojmenovic, Localized distance-sensitive service discovery in wireless sensor and actor networks, IEEE Transactions on Computers 58 (9) (2009) 1275–1288.
[56] Y.-C. Wang, W.-C. Peng, M.-H. Chang, Y.-C. Tseng, Exploring load-balance to dispatch mobile sensors in wireless sensor networks, in: IEEE Int'l Conf. Computer Comm. and Networks, 2007, pp. 669–674.
[57] S. Li, A. Zhan, X. Wu, P. Yang, G. Chen, Efficient emergency rescue navigation with wireless sensor networks, Journal of Information Science and Engineering 27 (1) (2011) 51–64.
[58] T.H. Cormen, Introduction to Algorithms, The MIT Press, 2001.
[59] E. Gelenbe, S. Timotheou, D. Nicholson, Fast distributed near-optimum assignment of assets to tasks, The Computer Journal 53 (9) (2010) 1360–1369.
[60] E. Gelenbe, Random neural networks with negative and positive signals and product form solution, Neural Computation 1 (4) (1989) 502–510.
[61] E. Gelenbe, S. Timotheou, Random neural networks with synchronised interactions, Neural Computation 20 (9) (2008) 2308–2324.
[62] Summit safety. http://www.summitsafetyinc.com/.
[63] J.-H. Huang, S. Amjad, S. Mishra, CenWits: a sensor-based loosely coupled search and rescue system using witnesses, in: ACM Int'l Conf. Embedded Networked Sensor Systems, 2005, pp. 180–191.
[64] G. Kantor, S. Singh, R. Peterson, D. Rus, A. Das, V. Kumar, G. Pereira, J. Spletzer, Distributed search and rescue with robot and sensor teams, Field and Service Robotics 24 (2006) 529–538.
[65] J. Wilson, V. Bhargava, A. Redfern, P. Wright, A wireless sensor network and incident command interface for urban firefighting, in: Int'l Conf. Mobile and Ubiquitous Systems: Computing, Networking and Services, 2007, pp. 1–7.
[66] Y.-D. Kim, Y.-G. Kim, S.-H. Lee, J.-H. Kang, J. An, Portable fire evacuation guide robot system, in: IEEE/RSJ Int'l Conf. Intelligent Robots and Systems, 2009, pp. 2789–2794.
[67] A. Ko, H.Y.K. Lau, Robot assisted emergency search and rescue system with a wireless sensor network, International Journal of Advanced Science and Technology 3 (2009).
[68] M. Klann, Tactical navigation support for firefighters: The LifeNet ad-hoc sensor-network and wearable system, Mobile Response, Lecture Notes in Computer Science 5424 (2009) 41–56.
[69] E. Gelenbe, A. Stafylopatis, Global behaviour of homogeneous random neural systems, Applied Mathematical Modelling 15 (10) (1991) 534–541.
[70] V. Atalay, E. Gelenbe, Parallel algorithm for colour texture generation using the random neural network model.
[71] E. Gelenbe, K.C. Sevcik, Analysis of update synchronisation algorithms for multiple copy data bases, IEEE Transactions on Computers C-28 (10) (1979) 737–747.
[72] E. Gelenbe, G. Hébrail, A probability model of uncertainty in data bases, in: ICDE, IEEE Computer Society, 1986, pp. 328–333.