2009-10-01

Building Fire Emergency Detection and Response Using Wireless Sensor Networks

Yuanyuan Zeng
University College Cork, yz2@cs.ucc.ie

Sean Og Murphy
University College Cork, jmm3@cs.ucc.ie

Lanny Sitanayah
University College Cork, ls3@cs.ucc.ie

See next page for additional authors

Recommended Citation

Zeng, Y., Murphy, S., & Sitanayah, L. (2009). Building fire emergency detection and response using wireless sensor networks. *Ninth IT and T Conference*, Technological University Dublin, Dublin, Ireland, 22nd-23rd October. doi:10.21427/905m-np47

Authors

Yuanyuan Zeng, Sean Og Murphy, Lanny Sitanayah, Tatiana Tabirca, Thuy Truong, Ken Brown, and Cormac Sreenan

Building Fire Emergency Detection and Response Using Wireless Sensor Networks

Yuanyuan Zeng, Seán Óg Murphy, Lanny Sitanayah, Tatiana Maria Tabirca, Thuy Truong, Ken Brown, Cormac J. Sreenan
Department of Computer Science, University College Cork
{yz2, jmm3, ls3, tabirca1, tt11, k.brown, cjs}@cs.ucc.ie

Abstract

Wireless sensor networks (WSNs) provide a low cost solution with respect to maintenance and installation and in particular, building refurbishment and retrofitting are easily accomplished via wireless technologies. Fire emergency detection and response for building environments is a novel application area for the deployment of wireless sensor networks. In such a critical environment, timely data acquisition, detection and response are needed for successful building automation. This paper presents an overview of our recent research activity in this area. Firstly we explain research on communication protocols that are suitable for this problem. Then we describe work on the use of WSNs to improve fire evacuation and navigation.

Keywords: Wireless Sensor Networks, Fire Detection, Fire Evacuation, Emergency Response

1 Introduction

In the near future we expect buildings to be equipped with a range of wireless sensors and actuators functioning as part of an overall building management system. Included in this set of sensors will be devices to monitor fire and smoke and to respond to the sensed events, allowing detection, localisation and tracking of fires, and providing guidance to evacuees and firefighters on the progress of the fire, on escape routes, and on the locations of people needing assistance. As part of the NEMBES project [1], we are developing a variety of techniques and application solutions to enable this vision of enhanced fire response through wireless embedded systems. In this paper, we present an overview of our work in two areas: protocol design for robust network operation, and sensor driven evacuation planning and simulation.

The remainder of this paper is structured as follows: Section 2 presents the routing and MAC layer designed especially for building fire. Section 3, we outline the evacuation and guidance in fire, and then propose the emergency simulation. Section 4 involves some related work. Finally, Section 5 concludes this paper.

2 Routing and MAC Layer Design for Building Fire

Wireless sensor networks for sensing and reporting on a spreading fire are faced with two main issues. Firstly, large volumes of data need to be reported as quickly as possible to a central sink (also called base station) – the rate of sensing will be greatly increased over normal operation, requiring more frequent data transmission. Access protocols and schedules used during normal conditions will no longer apply; instead new protocols designed to ensure rapid transmission of critical data without increased collisions are required. Secondly, the network itself will degrade as the fire spreads, blocking links and killing individual nodes. Stored routing information will quickly become invalid, and whole areas of the network may become disconnected. Adaptive routing protocols are required which can adapt quickly to the changing network, which can act opportunistically, and which are robust to the spreading fire. On the other hand, energy efficiency and node lifetimes are of little concern. We investigate three techniques for operation of an in-building sensor network during a fire: real-time robust routing, a routing protocol able to take advantage of transient connectivity provided by firefighters, and traffic-adaptive MAC. We present each of these in turn below.

2.1 Real-time and Robust Routing in Fire (RTRR)

RTRR is the core routing protocol that we have developed for use in building emergency networks. Its key requirement is to deliver messages in real-time and with a high probability of success which is the main challenge in building fire emergency. To achieve this, it employs the use of several techniques. Firstly, it maintains delay estimates from each node to its nearest sink to guide a real-time delivery. Secondly, it tracks the status of nodes and link valid time in fire, allowing traffic to avoid nodes that are in danger according to fire spreading. Thirdly, it uses adaptive transmission power to avoid routing holes caused by nodes that have failed or seek real-time and valid paths in fire situations.

Given a WSN with N sensors and M sinks deployed in a building, with a goal of each sensor being able to deliver its data packets to one of the sinks within maximum delay T_{max}. Each sensor can adjust its transmission range by using different transmission power levels $p_0, p_1 \ldots p_{k-1}=p_{max}$. Initially, all sensors transmit at default power p_0. Nodes maintain information on their route to the sink and on their immediate neighbourhood. Each node is in one of four states: *safe*(no fire), *lowsafe*(1-hop to fire), *infire*(caught in fire) and *unsafe*(cannot work). A node may change its state autonomously in response to tracked fire situations: occurrence, expanding, diminishing, etc..

Each sink periodically broadcasts a HEIGHT message to refresh the network, allowing nodes to determine reachability to the nearest sink with “height” (defined as number of hops toward the nearest sink) and estimate delay. We denote *delay* (*sink*, *i*) as the delay experienced from the sink to node *i*, and then we use *delay* (*sink*, *i*) as a bound to guide a real-time delivery from node *i* to sink. The estimate delay is calculated by cumulative hop-to-hop delay:

$$delay(\text{sink}, i) = \sum_{n=1}^{h} Avg\_delay = \sum_{n=1}^{h} (T_c + T_t + T_q) * R \tag{1}$$

In formula (1), n is the hop count from the sink to node i; T_c is the time it takes for each hop to obtain the wireless channel with carrier sense delay and backoff delay. T_t is the time to transmit the packet. T_q is the queuing delay, and R is the retransmission count. The *delay* (*sink*, *i*) is a bound to guide the real-time forwarding [12]. Furthermore, we can provide a good estimation of the delay by adjusting it based on both the weighted average and variation of the estimated variable

Based on this, each node selects the relay based on metric with height, estimate delay and node state as follows:

(1)Firstly, filter to find the nodes with lower height than current node.

(2)Secondly, select the node with enough slack time (defined as time left) compared to estimate delay.

(3)Thirdly, we filter the remaining forwarding choices by node state in the priority from “*safe*” to “*infire*”.

(4)If there is more than one node satisfied, we select the relay with the higher residual energy. If there is still a tie, we choose the lower ID.

If no suitable relay is found, the node increases its power level gradually to find another existing neighbour or invoke a new neighbour discovery, and try to jump over the hole.

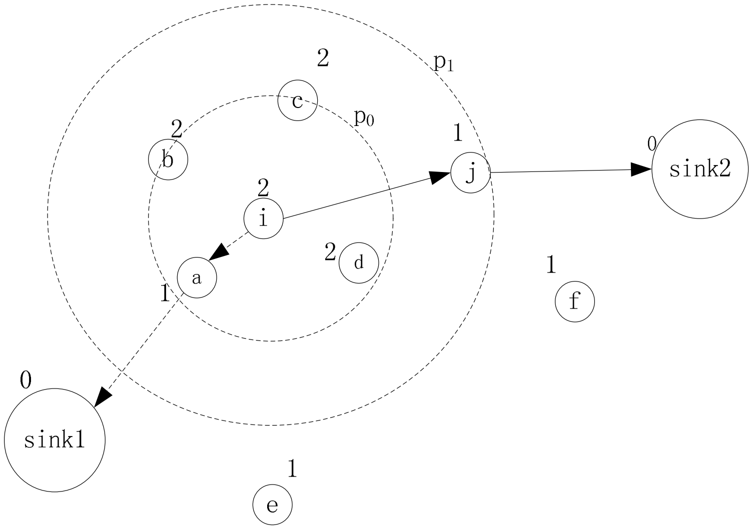

Fig.1 Increase power to jump over the hole

Fig.1 shows the new neighbor discovery. The sink*1* and *sink2* are two sinks, and the other nodes are sensors. The number beside each node represents the "height" of each node toward the sink. Node *i* reports and routes the data to the sink. The path: {*i*, *e*, sink*1*} (with p_0 from sensor *i* to *e*) is invalid because slack does not satisfy the estimated end-to-end delay. If there are no existing eligible neighbours, then *i* will increase its power to p_1 to reach node *j* and delivers the packets to another sink: *sink2* by path {*i*, *j*, *sink2*} when slack on this route is no less than delay estimation.

In building fire emergencies, robust routing is crucial due to the impact of quickly moving fire on node liveness. We assume that: (1) the minimal time interval between "*infire*" and "*unsafe*" state of a node is chosen as a parameter known beforehand. (2) We use necessary transmission range for connectivity between nodes (according to selected power level) to approximate the minimal fire spreading time between two nodes. In practice there are well-known guidelines for estimating the rate of fire spread, taking into account building materials, etc. It's also the case that obstacles, such as walls, that mitigate radio propagation also have the effect of slowing fire spread.

When a relay is used for routing, we add a *timeout* to avoid the use of stale and unsafe nodes, i.e., every node on the path from source to destination has a *timeout* to record the valid time. At the same time, each link's valid time is decided by the nodes adjacent to it. The *timeout* is updated when node state changes among the neighbourhood. The relay and its adjacent path links that exceed the *timeout* value is considered invalid and then evicted. Accordingly, a routing re-discovery is invoked to find another relay with a valid route path onward one of the sinks (may be a different sink from current one).

2.2 Opportunistic Routing With Mobile Sinks

We now consider scenarios where the network is damaged: routes to the sink may be very long for some nodes, and other areas are now completely disconnected. We envisage firefighters entering the building with small specialized sensor nodes attached to them. These nodes can act as mobile sink nodes, able to relay data back to the main static sink in a single hop, and so provide new transient paths to the static sink. We assume, though, that the firefighters are concerned only with fire fighting and rescue, and thus network issues have no influence on the movement of the mobile sinks. The main question we consider is how to make best use of these mobile sinks. When should sensor nodes relay data via the mobile sink? How does the mobile sink make its presence known to the sensor nodes? How can we use the mobile sink to re-connect disconnected regions of the field? We assume an underlying routing protocol for the network similar to RTRR. Thus each node maintains information on its relay node and hop count for transmitting data to the static sink through the network. First, we assume that the mobile sink transmits a beacon as its moves through the building. If the speed of movement is higher than a threshold, the beacon signal is suspended. Nodes that receive the beacon forward it for up to k hops. Each node then decides whether or not to use this new transient route. Each node, however, also maintains its old route. When the mobile sink moves out of range, the links to it will be broken, and the nodes revert to their old routes. Secondly, we assume that nodes in a disconnected region reply to the beacon with a panic code, which causes the mobile sink to change its beacon to indicate that it will only relay data from the disconnected region. This gives priority to the disconnected region to transmit whatever buffered data it has been able to store. Thirdly, we envisage the mobile sink using a directional antenna to transmit predictive beacons announcing its expected arrival, assuming it maintains its current speed and trajectory. Nodes receiving the predictive beacon can then decide whether to buffer data and wait for the arrival of the mobile sink.

In the first and third cases, the main issue is in the tradeoff between taking advantage of the newly available shorter routes and wasting time transmitting control messages and rerouting data only to find that the mobile sink has moved on and is no longer available. If the behaviour is too conservative, opportunities to transmit data are lost; if the behaviour is too aggressive, latency increases and data is lost as the new routes disappear while data is in transit.

2.3 A Hybrid MAC Protocol for Emergency Response (ER-MAC)

During an emergency situation, sensor nodes must be able to adapt to a very large volume of traffic and collisions due to simultaneous transmissions. Nodes must accurately deliver the important information to the sink in no time. Moreover, in this emergency situation, energy efficiency of the communication protocol can be traded for the necessity of high throughput and low latency. In WSNs, Medium Access Control (MAC) plays an important role in a successful communication.

We design ER-MAC, a hybrid MAC protocol for fire emergency. This protocol adopts TDMA approach to schedule collision free transmission toward the sink. During normal day-to-day monitoring, the communication is delay-tolerant and must be energy efficient to prolong the network lifetime. Therefore, each node only wakes up to transmit and receive messages according to its specified schedule. It, otherwise, sleeps to conserve energy. When an emergency event occurs, the nodes change the behaviour of the MAC by allowing contention in TDMA slots. A node may contend for its neighbour's transmit slot if it has priority packets to send. Furthermore, during an emergency situation, all nodes wake up at the beginning of each TDMA slot for possible reception of packets. Our MAC protocol uses a pair of priority queues as shown in Fig.2 to separate two types of packets, i.e. high priority packets and low priority packets. The rule is low priority packets are sent if the high priority queue is empty. Inside a queue, packets are ordered based on their *slack*, that is the time remaining until the packet deadline expires.

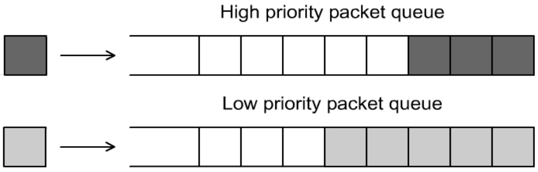

Fig.2 Priority queues

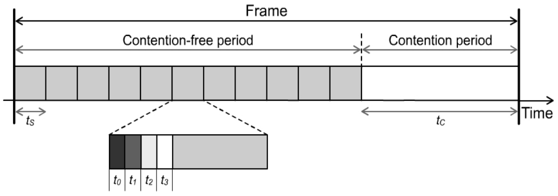

Fig.3 Frame structure of ER-MAC

Fig.3 shows a frame structure of ER-MAC, which consists of contention-free slots with duration t_S each and a contention period with duration t_C. In each contention-free slot, there are sub slots t_0, t_1, t_2 and t_3 for contention that will be explained below. Note that the period of $t_S - (t_0 + t_1 + t_2 + t_3)$ is sufficient to carry a packet. We include a contention period at the end of each frame to support addition of new nodes. During the no fire condition, every node sends its own data and forwards its descendants' data to its parent in collision-free slots. A node has a special slot to broadcast synchronization message to its children. However, as soon as the fire alarm is triggered, node changes the behaviour of MAC as follows:

(1) An owner of a slot wakes up in the beginning of its own transmit slot. If it has a high priority packet to send, it transmits the packet immediately. If the owner has no high priority packet to send, it allows its one-hop neighbours with high priority packets to contend for the slot.

(2) All non-owners of the slot wake up in the beginning of every slot to listen to the channel for possible contention or reception of packets. If a non-owner with a high priority packet senses no activities in the channel during t_0, it contends for the slot during t_1. The owner of the slot replies the requester’s request.

(3) The owner of the slot with low priority packets can only use its own slot if during $t_0 + t_1$ it does not receive any slot request messages from its neighbours.

(4) A non-owner with low priority packet can contend for the slot if during $t_0 + t_1 + t_2$ it senses no activities in the channel. It then contends for the slot during t_3 and the owner of the slot replies to the requester’s request.

3 Fire Evacuation and Navigation

Our main application is navigation guidance for both firefighters and evacuees. We assume two families of sensors, one able to report on the numbers and locations of people in the building and one able to report on the current extent and state of the fire. We also assume access to the building plans from which, combined with sensed data, we can compute the predicted spread of the fire and compute the quality of navigation paths through the building. We are developing algorithms for computing safe and short paths from each location to designated exits and for updating these paths as new sensed data arrive. We are also constructing a simulation framework in which we are able to simulate the actuation of navigation signs and the movement of people as they attempt to follow the signs and evacuate the building.

3.1 Evacuation path planning

The core of core of our approach is represented by a dynamic model for fire hazard spreading in building environments. The dynamic model provides estimated information about the dynamicity of the fire hazard over time in the building environment. The model then generates a set dynamic navigation weights $c^{(t)}(u,v)$ representing the time taken to walk between two adjacent locations *u, v* at the time *t*. Based on these elements two types of dynamic navigation paths are introduced within the building environment. Firstly, the dynamic shortest paths are considered to be used by well-able evacuees towards the exit or by the fire-fighters to navigate in the building. The second type of path uses the concept of safety which represents the maximum time one can safely delay at the nodes. These dynamic safety paths can be used in evacuation by evacuees with disability of by fire-fighters assisting injured evacuees. The dynamic model also generates a series of dynamic centrality indices that offer valuable information about the importance of each node in the evacuation process. Perhaps, the most important index is represented by the dynamic betweenness which gives the probability of a node to be on evacuation paths.

The first scenario is for evacuation and it is based on a centralised computation. The WSN network senses the hazard locations and then notifies the sink node about them. At the sink node the dynamic model is simulated and estimated information about the hazard development, about the dynamic shortest paths and about the dynamic safety paths are generated for future time. Then this information is transmitted from the based station to the actuator sensors which can display the best or safest route to take. This approach offers always accurate evacuation data and avoids the WSN network becoming congested by the process of updating evacuation routes. Another approach of this scenario is when the estimated evacuation information is sent from the sink node to the fire-fighters in order to allow them use only safe navigation routes to the exit. The second scenario uses the dynamic model to offer the fire-fighters support when they navigate in the building. An important duty of fire-fighters is to search rooms for possible injured people and to assist them in evacuation. In this case the fire-fighters use the dynamic shortest paths in the navigation process through the rooms and then take the dynamic safety path to the exit when they assist injured evacuees. The third scenario offers information to the Incident Commander about the most important nodes in evacuation which should be kept hazard free during the evacuation process.

3.2 Multi-Agent Emergency Simulation

We design a real-time simulator for detecting and handling building fire emergency scenarios. The goals of this simulator are to provide for: (1)a dynamic virtual test-bed for population routing and networking algorithms during emergencies, (2)identification of building features that impact on evacuation scenarios, such as corridors prone to congestion, (3)visualising real-world emergency situations and predicting outcomes to inform rescue personnel as to the best rescue strategy or possible danger areas.

The underlying world model for this simulation is an object-based 2.5 dimension "building". Each floor of the building is a 2D collection of world objects, with the floors arranged in a spacial collection (ground floor, first floor, second floor etc). Stairs, fire escapes and elevators provide a mechanism for agents to travel between floors. This 2-and-a-half dimension model was chosen as it simplifies agent behaviour computations and allows for very clear visualisation of the emergency as it unfolds. The underlying building objects have analogues within the Industry Foundation Classes building model objects, such as walls, doors and so on.

The simulation features multiple agents with dynamic behaviours navigating a building during an emergency. These agents are driven by a Sense->Plan->Act cycle and have basic memory. The two main classes of Agent are "Occupant" agents (persons present in the building, primarily driven by environmental cues such as direction signs or following crowds) and "Firefighter" agents (primarily driven by individual instructions, such as radio contact or personal "compass" direction). Agents will have steering and crowding mechanisms to accurately reflect real-life population movement. The underlying physical model of the world combined with such measures will provide useful knowledge as to areas in the building with excessive traffic and poor movement flow, or parts of a building which are of high-importance for evacuation (e.g. a main corridor).

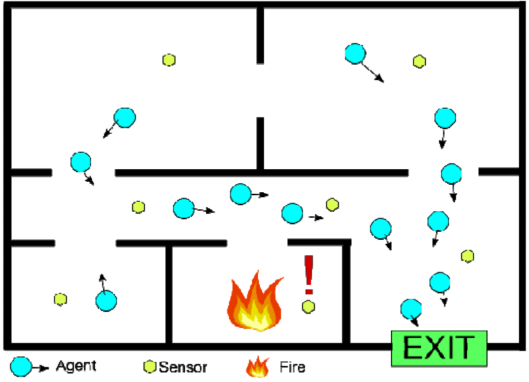

Fig.4 Simulation illustration

The simulation also incorporates simulated embedded network elements. These virtual sensors detect people, fire, smoke and temperature. The simulated actuators will drive building elements such as direction signs, windows, door locks and fire suppression systems (sprinklers etc). Fig. 4 shows a screenshot of our simulation for building fire.

The sensors will be used to drive a view of the building apart from the actual underlying simulation itself. This "sensor view" is limited by sensor uncertainty, sensing range characteristics and sensing schedules. This limited view of the building provides information to a higher-level Application Layer which will be running Evacuation route planning algorithms, fire-fighter direction and other emergency applications.

The systems running on the application layer feed actuation instructions to the in-simulation actuators which reflect these instructions in the underlying simulation (signs direct the occupants along the evacuation path, sprinklers activate, fire-fighters remotely receive a new instruction, and so on).

4 Related work

Our research discussed in this paper is based on the NEMBES project funded by the Irish Higher Education Authority under the PRTLI-IV programme. NEMBES is an inter-institutional and multi-disciplinary research programme that will investigate a "whole system" approach to the design of networked embedded systems, marrying expertise in hardware, software and networking with the design and management of built environments. Our research is covered by one of the main research strands in NEMBES: facilities management as “sensor network management within buildings”. The focus of the research is to develop dynamic sensor network management methodologies for building environment where wireless sensor network technology providing low cost data acquisition also provides a means of detecting the environment and the combine wireless sensing and actuating capabilities to provide some response capability for sensed events. While the network routing and MAC protocols govern the successful data reporting of the wireless sensor network, it will also be tasked with fire events via alarm triggers. These alarmed events can be interpreted, ranked and routed based on urgency and maintenance, repair, replace requests or highlight the need for additional equipment/sensors meters to satisfy building services demands such as making fire evacuation for people in fire and providing guidance for firefighter to find injured.

There are a lot of routing and MAC layer protocols designed for WSNs. Real-time design is one of the challenges in building fire emergency. Some WSN applications require real-time communication, typically for timely surveillance or tracking, e.g. SPEED [2], MM-SPEED [3], RPAR [4] and RTLD [5] were designed for real-time applications. But they are not well suited for building fire emergency especially the situation will be even worse with dynamic topology changes and node failure caused by fire spreading.

In building fire emergency applications, we envisage firefighters entering the building with small base stations attached to them. These base stations can act as mobile sink nodes, able to relay data back to the main base station in a single hop. Recently, many researchers have considered mobile relays or mobile sinks to solve the sink neighbourhood problem [10, 11]. In these scenarios, mobile nodes play an important role for relaying or collecting data continuously. Combining our application, the main question we consider in fire is how to make best use of these mobile sinks.

In WSNs, Medium Access Control (MAC) plays an important role in a successful communication. Existing contention-based MAC protocols such as S-MAC [6], schedule-based MAC protocols such as TRAMA [7], and the combination of both contention and schedule (hybrid) for example Z-MAC [8] are not suitable for fire emergency. During this emergency situation, successful communication of the WSN depends on a robust and reliable communication protocol to transport important messages to the base station. Furthermore, in the emergency situation, energy efficiency of the communication protocol can be traded for the necessity of high throughput and low latency. Different from existing work, nodes change the behaviour of the MAC by allowing contention in TDMA slots when an emergency event occurs. A node may contend for its neighbour's transmission slot if it has priority packets to send.

The last couple of years have seen an important number of applications of sensors in building environments. The usage of the WSN networks in emergency evacuation is just one of them with various solutions proposed so far [9]. Different from this, our work uses a novel dynamic evacuation model [13] to consider dynamic evacuation graph with fire spreading.

Currently, there is no simulator that is designed specifically for emergency applications such as building fire. We designed a simulator that could provide a dynamic virtual testbed for designed protocols and algorithms especially for emergency scenarios.

5 Conclusions

In this paper, we outline some of the main ideas of our NEMBES project work on building fire emergency applications. Firstly, we present the mechanism of the real-time and reliable routing protocol designed for building fire to guarantee a delay bounded and high successful probabilistic end-to-end data delivery in fire. Secondly, we propose an opportunistic routing scheme with mobile sinks. Thirdly, we present a MAC protocol that is adaptive to priority-based traffic and collisions due to simultaneous transmissions. Next, we give some details about fire evacuation/navigation mechanism by using a dynamic evacuation model. At last, we bring forward a simulation testbed especially for building fire based on the protocols we designed.

Our research is still in progress and it could benefit applications for building fire emergency and other similar emergency situations such as earthquakes and other urban disasters. The further work includes exploring the complementary of existing protocols and mechanisms, as well as implementing simulations under different network scenarios and fire models.

References

[1] Networked Embedded Systems (NEMBES), http://www.nembes.org.

[2] T. He, J. Stankovic, C. Lu, and T. Abdelzaher, “SPEED: A Stateless Protocol for Real-time Communication in Sensor Networks,” *23rd International Conference on Distributed Computing Systems*, pp.46-55, May. 2003.

[3] E. Felemban, C. –G. Lee, E. Ekici, R. Boder, and S. Vural, “Probabilistic QoS Guarantee in Reliability and Timeliness Domains in Wireless Sensor Networks,” *24th Annual Joint Conference of the IEEE Computer and Communication Societies(InfoCom’05)*, 4(4), pp. 2646- 2657, Mar 2005.

[4] O. Chipara, Z. He, G. Xing, Q. Chen, et. al., “Real-time power-aware routing in sensor networks,” *14th IEEE International Workshop on Quality of Service*, pp.83-92, 2006.

[5] A. Ahmed, N. Fisal, “A real-time routing protocol with load distribution in wireless sensor networks,” *Computer Communications*, 31(14), pp.3190-3203, 2008.

[6] W. Ye, J. Heidemann, and D. Estrin, “An Energy-Efficient MAC Protocol for Wireless Sensor Networks”, *21st Annual Joint Conference of IEEE Computer and Communications Societies (InfoCom’02)*, 3(3), pp.1567- 1576, Jun.2002.

[7] V. Rajendran, K. Obraczka, and J.J. Garcia-Luna-Aceves, “Energy-Efficient Collision-Free Medium Access Control for Wireless Sensor Networks”, *1st International Conference on Embedded Networked Sensor Systems (SenSys’03),* pp.181-192, Nov.2003.

[8] I. Rhee, A. Warrier, M. Aia, and J. Min, “Z-MAC: A Hybrid MAC for Wireless Sensor Networks”, *3rd International Conference on Embedded Networked Sensor Systems (SenSys’05)*, pp.90-101, Nov.2005.

[9] M. Barnes, H. Leather and D.K. Arvind, “Emergency Evacuation using Wireless Sensor Networks”, *32nd IEEE Conference on Local Computer Networks (LCN’07),* pp 851-857, Oct. 2007.

[10] Yihong Wu, Lin Zhang, Yiqun Wu, Zhisheng Niu, “Interest Dissemination with Directional Antennas for Wireless Sensor Networks with Mobile Sinks,” *4th International Conference on Embedded Networked Sensor Systems (Sensys’06)*, pp.99-111, 2006.

[11] Wook Hyun Kwon, Tarek F. Abdelzaher, Hyung Seok Kim, “Minimum-Energy Asynchronous Dissemination to Mobile Sinks in Wireless Sensor Networks,” *1st International Conference on Embedded Networked Sensor Systems (SenSys’ 03)*, pp.193-204, Oct.2003.

[12] Yuanyuan Zeng, Cormac Sreenan, “A Real-Time and Robust Routing Protocol for Building Fire Emergency Applications Using Wireless Sensor Networks”, *8th International Workshop on Real-Time Networks(RTN'09)*, 2009.

[13] Tatiana Tabirca, Kenneth N. Brown and Cormac J. Sreenan, “A Dynamic Model for Fire Emergency Evacuation Based on Wireless Sensor Networks”, *8th International Symposium on Parallel and Distributed Computing (ISPDC),* July 2009.