

Sensor Network Navigation without Locations

Jiliang Wang, *Member, IEEE*, Zhenjiang Li, *Member, IEEE*, Mo Li, *Member, IEEE*, Yunhao Liu, *Senior Member, IEEE*, and Zheng Yang, *Member, IEEE*

Abstract—We propose a pervasive usage of the sensor network infrastructure as a cyber-physical system for navigating internal users in locations of potential danger. Our proposed application differs from previous work in that they typically treat the sensor network as a media of data acquisition while in our navigation application, in-situ interactions between users and sensors become ubiquitous. In addition, human safety and time factors are critical to the success of our objective. Without any preknowledge of user and sensor locations, the design of an effective and efficient navigation protocol faces nontrivial challenges. We propose to embed a road map system in the sensor network without location information so as to provide users navigating routes with guaranteed safety. We accordingly design efficient road map updating mechanisms to rebuild the road map in the event of changes in dangerous areas. In this navigation system, each user only issues local queries to obtain their navigation route. The system is highly scalable for supporting multiple users simultaneously. We implement a prototype system with 36 TelosB motes to validate the effectiveness of this design. We further conduct comprehensive and large-scale simulations to examine the efficiency and scalability of the proposed approach under various environmental dynamics.

Index Terms—Wireless sensor networks, navigation, location-free, road map

1 INTRODUCTION

RECENT advances in wireless sensor network (WSN) technologies provide us with the ability to offer pervasive usage of sensor networks widely deployed over the space of interest. The increasing study of WSNs aims at enabling computers to better serve people by automatically monitoring and interacting with the physical world [1], [2], [3], [4]. Existing works, however, largely focus on developing sensor network systems principally providing remote data collection. The possibility of in-situ interactions between the users and their physical environment is overlooked. Such interactions could significantly expand the capability of WSNs and thereby enhance their usability.

This work proposes to utilize the sensor network infrastructure as a cyber-physical system for navigating internal users during emergencies. The users are equipped with communicating devices like 802.15.4 compatible PDAs to communicate with sensors in the network. In the event of an emergency, the sensor network explores the emergent field and provides the necessary guidance information to navigate the user to safety.

The proposed application essentially differs from prior works in several aspects. First, most of previous works view the sensor network as a mechanism for data

acquisition, concentrating on organizing a data-centric network for efficiently collecting, routing, processing in-network sensory data [5], [6], [7], [8], [9] and the like. In contrast, our application focuses more on in-situ user interaction with the sensor network infrastructure. There are not necessarily one or more sinks as data processing centers, yet there are no needs collecting back the sensory data distributed over the field. All operations are in situ carried out by cyber-physical interactions among users and network nodes. Second, the navigation of human beings is inherently different from routing data packets. We have various methods to compensate for network issues like packet loss, for example, packet reroute, multipath routing, data redundancy and so on. The navigation of human beings, however, is the safety-critical selection of a single route, which prevents us from simply borrowing existing packet routing protocols. For example, the recently suggested opportunistic routing paradigm, though it provides extra efficiency for the delivery of wireless data, obviously cannot be applied to design navigation protocols since it is impossible for human beings to be physically multicast or copied in their movement. Third, the time factor becomes critical in the context of human navigation other than the data delivery in the network. The limited human movement speed dictates that the navigation process is time consuming, while emergency situations might result in time variation. As emergency or dangerous situations change, it becomes necessary to frequently update the route plans for the guided users. Path dynamics in traditional packet routing process typically occurs between delivering different packets, while such dynamics in the navigation process might exist all along for guiding single individuals. People may even move backward to seek broader opportunities under varied situations.

There have been attempts made at guiding navigation using WSNs. Most, if not all, existing approaches assume the availability of locations on each sensor node. Knowing

- J. Wang, Y. Liu, and Z. Yang are with the School of Software, Tsinghua University, 11-313, Dong Pei Lou, Haidian District, Beijing, P.R. China. E-mail: {jiliang, yunhao, yang}@greenorbs.com.
- Z. Li is with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Room 3528, Clear Water Bay, Kowloon, Hong Kong. E-mail: lzjiang@cse.ust.hk.
- M. Li is with the School of Computer Engineering, Nanyang Technological University, N4-02C-108, 50 Nanyang Avenue, Singapore 639798. E-mail: limo@ntu.edu.sg.

Manuscript received 21 Dec. 2011; revised 11 June 2012; accepted 22 June 2012; published online 2 July 2012.

Recommended for acceptance by W. Jia.

For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2011-12-0921. Digital Object Identifier no. 10.1109/TPDS.2012.207.

the locations of dangerous areas, the sensor network can perform easy and efficient route calculations to navigate internal users out of the emergency area. The location information, however, may not always be available in many realistic situations where emergency guidance are needed, for example, in an underground tunnel or coal mine, a complex indoor area, and so on. The requirement of location information largely constrains the applicability of existing approaches to location-free environments. In addition, existing approaches usually do not specifically consider the impact of variations of dangerous areas, for example, the expansion, shrinking, or disappearing of areas which is deemed dangerous. In reality, such variations often degrade the effectiveness of existing designs or even overwhelm them.

The design objectives of this work are twofold. First, to release the necessity of utilizing location information. Specifically, neither the sensor nodes nor the users need to know their instant locations to achieve successful navigation. Second, to address emergency dynamics that can lead to variations of dangerous areas. Neither of the above is fully addressed in existing literatures. In this work, we propose to embed a distributed road map across the sensor network, which performs as a public infrastructure for providing guidance information for internal inquirers. Users then would be able to issue queries from the road map and follow the recommended routes, avoiding dangerous areas. We also design efficient mechanisms to update the road map system according to the variations of dangerous areas thereby maintaining its accuracy and effectiveness. We implement a prototype system with 36 TelosB Motes as well as conduct large scale simulations to validate the scalability and examine the performance of our design. Experimental results show that this approach is effective and highly scalable when the network size becomes large and multiple users are simultaneously navigated in the network.

The remainder of the paper is organized as follows: Section 2 presents problem specification. Section 3 describes our design principles. We present our implementation experience in Section 4. We further validate this design with experiments on our prototype system. We evaluate the scalability of our approach with large-scale trace driven simulations in Section 5. We discuss the related work in Section 6 and conclude the work in Section 7.

2 PROBLEM SPECIFICATION

We consider the scenario of navigating human beings on the field under emergencies, where there might be several dangerous areas that threaten the safety of human beings, for example, excessive heat, poisonous gas, passage obstacles and so on. People need to be guided out of the field while keeping away from those dangerous areas. Fig. 1 depicts such an example scenario. We characterize the navigation problem as a path planning problem and present its assumptions, objectives, and requirements as follows.

Assumptions. We assume an emergent field containing several areas of dangers, as the red areas shown in Fig. 1. The dangerous areas might emerge, disappear, expand, or

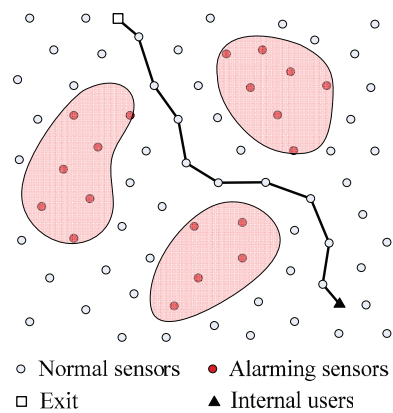


Fig. 1. Sensor network navigation.

shrink as the time passes. The number of dangerous areas at any time is finite.

A sensor network system is deployed in the field, where each sensor is able to detect the dangers distributed over the field. The sensor node triggers a “yes” alarm if it resides in the dangerous area (red nodes in Fig. 1) and triggers “no” if outside (sky-blue nodes in Fig. 1). Thus, the boundary of a dangerous area can be outlined by the pairs of neighboring sensors with different outcomes. Each user carries a communicating device like 802.15.4 compatible PDA that can talk with sensors. By measuring the strength and direction of wireless signals, the user is able to track any targeted sensor node [10]. Thus, the navigating route can be interpreted as a sequence of nodes.

Objectives. The objective of a successful navigation is to plan a path for each user to one or more preknown exits on the field that lead to safe departure, bypassing all the dangerous areas. In Fig. 1, there is an exit in the fields that users are required to lead to. We depict such an example route that leads the internal user to the exit. The navigation process is carried out in a fully distributed manner without any dedicated central agents like sinks. Each user is hand-off guided by sensors along the entire route.

Requirements. We mainly have the following three requirements on the navigation protocol:

- We require that the selected navigation route is *safe*, i.e., the route should be apart from the dangerous areas with guaranteed safety.
- We require that the selected navigation route is *efficient*, i.e., the route should not be excessively long. A shorter route results in a quicker departure from dangers.
- We require that the navigation protocol is *scalable*, i.e., the building and updating of the navigation routes should be local and lightweight.

3 DESIGN PRINCIPLES

We elaborate the design of our navigation system in this section. The main idea is that we embed a distributed road map system in the sensor network. This road map system is built according to the distribution of dangerous areas and thus can characterize the features of the safety in the field. The navigation system maintains the road map as a public

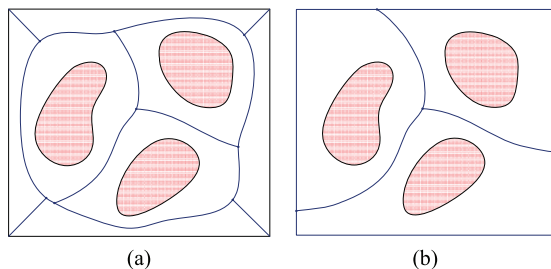


Fig. 2. The basic road map framework.

infrastructure across the network and guide different users across the field through the same road map, saving unnecessary overhead of individually planning routes for different users. The road map is updated in an event-driven manner when the dangerous areas vary.

In the following, we in detail describe the design principles in four components: *building the road map*, *guiding navigation on the road map*, *reacting to emergency dynamics*, and *improving route efficiency*. In each component, we illustrate the design principles in continuous settings and describe their properties accordingly.

3.1 Building the Road Map

We denote the entire emergent field as region E and the combination of dangerous areas as region D . Thus, the road map is built in the remainder region $R = E \setminus D$, since human beings can only move outside dangerous areas for ensuring their safety.

We build the basic framework of the road map by concatenating the medial axis of region R . The medial axis is a set of points, each of which is closest to at least two different points on the boundaries of dangerous areas. Fig. 2 shows the basic framework of the road map (blue lines) on the continuous field of dangerous areas. We treat the area out of the sensor field as dangerous, since without any sensing information about such area we have to consider it possibly dangerous.

The consequent road map is then built as Fig. 2a depicts. We can also choose to consider the sensor field boundary safe with some preliminary information on the boundary, for example, the sensor field is indoor environment, safely surrounded by walls or fences. In this case, the road map is built as Fig. 2b depicts. Since the two cases are essentially similar, without loss of generality, in the following, we mainly focus on the first case.

As proven in [11], the medial axis of region R is a finite set of continuous curves and it retains the topological features of this region. Thus, our road map framework is expressive, which captures the topological features of the safe region R , representing the possible safe corridors among dangerous areas with curve segments on the medial axis. The road map framework is also compact, which represents the topological and geometric features of region R by a simple curve graph, the size of which is proportional to the complexity of large geometric and topological features on R [11].

3.2 Guiding Navigation on the Road Map

We utilize the road map framework as a backbone for navigating different users inside the field. The road map divides region R into different cells. Each cell is separated by road segments from others and contains a dangerous area inside it.

3.2.1 Connecting the Exit to the Road Map Backbone

First, we find the exit in one of the cells and build a route connecting the exit and the road map backbone. The route is calculated based on the distance from the boundary of dangerous area inside that cell. The basic procedure to detect the boundary of a dangerous area is as follows. First, based on the sensory data, sensor nodes can determine whether they are in a dangerous area. For instance, the temperature data from a sensor can indicate whether the sensor node is in a fire or not. Then, by locally exchanging such information, a sensor node can be aware of whether it is a boundary node. If a sensor node itself is in a fire and its neighbor is outside of the fire, this sensor node can infer that it is a boundary node. The real-world RF effects may cause some detection errors. However, the granularity of the introduced error is normally in the order of several hops. The road map is always built on sensors outside the dangerous areas. Therefore, the road map is still safe and can guide people to the exit of the dangerous field in practice. Thus, the path in the navigation system is still safe.

Then, we assign a virtual power field in the cell, where the power P of each point is inversely proportional to its distance d from the dangerous area, for example, $P = 1/d$. The route from the exit extends at each point along the most descending direction of the virtual field until reaching the road backbone.

We prove in Appendix A, which can be found on the Computer Society Digital Library at <http://doi.ieeecomputersociety.org/10.1109/TPDS.2012.207>, that the local minimum points of the virtual power field in each cell only reside on the medial axis. The proof guarantees that we can successfully build a route connecting the exit to the road backbone without being halted at any local minimum inside the cell. Such a route ensures that any point on the route is not further closer to the dangerous area than the destination exit. The route connecting the exit to the road backbone at a point that we call *gateway* in this paper, which can be treated as an exit on the road backbone. In [12], the flow complex has been proposed to achieve a similar goal. It requires a central controller to frequently collect information from the network and further conduct intensive computation, which is not suitable for a large-scale navigation system.

3.2.2 Assigning Directions on the Road Map

On the road backbone, we accordingly assign directions for each road segment, forming a safe path toward the gateway from each point on the backbone.

This can be achieved by flooding from the gateway throughout the road backbone. The flooded information includes the closest distance to the dangerous areas, d_c , along the road to the gateway, the path length along the road to the gateway, d_r , and the direction D along the road. Each point receives the flooded information from

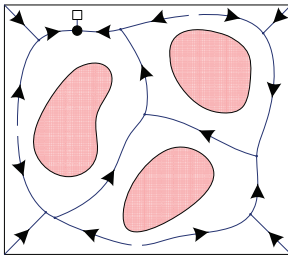


Fig. 3. The finally obtained directional road backbone.

different directions. Usually, the flooded information only comes from the two directions along the road but possibly multiple directions at the branch points where multiple roads segments intersect. Each point first compares d_c and maintains the path with the largest d_c . Among the paths with the same value of d_c , the point keeps the shortest path with the smallest d_r . Each node records the direction D of such a path. Finally, each point knows a path toward the gateway and maintains the direction of this path. Fig. 3 depicts the finally maintained directions on the road map. If we break from the end of each path, the finally obtained directional road backbone is a tree rooted at the gateway. We show in Appendix B, available in the online supplemental material, that there is no loop after the direction assignment. Lemma 1 shows the safety of the backbone.

Lemma 1. *On the road backbone, from any point to the gateway, the path along the assigned directions maximizes the minimum distance to the dangerous areas.*

Proof. Based on the direction assignment, each point maintains the direction to the gateway with the largest d_c . If there are multiple paths, the same minimum distance to the dangerous areas, the point keeps the shortest one. Thus, the final assigned direction must point to the path that maximizes the minimum distance to the dangerous areas. \square

3.2.3 Exploring the Routes for Users

As aforementioned, the road backbone divides the region into different cells. Each user initially resides in one of those cells. Navigating each user to the destination exit includes three stages.

In the first stage, each user is guided from the inside of the cell to the road backbone. The route is calculated similarly as the route connecting the exit and the road map. By assuming a virtual field around the dangerous area in the cell, where the power of each point is inversely proportional to its distance from the dangerous area, the user at each step moves along the most descending direction of the virtual field until reaching the road map backbone.

Along the road backbone, the route is selected simply according to the directions assigned on the road map, i.e., each user moves along the directional roads. The last-mile navigation is guided along the route that connects the exit and the gateway on the road map.

So far, our approach has been designed based on the consistence of radio signal and human beings, in which we assume that if there is a radio link, human beings can go through. However, our approach can also be applied to

other scenarios. We can combine the connectivity map with an actual physical map so as to construct a road map where all paths can be passed through by human beings. As a matter of fact, there have been approaches proposed to build the *mobility graph* [13] with paths that people can go through. In [13], after the deployment, by learning the moving trace of human beings in the network, the proposed algorithm can intelligently learn between which pair of sensor nodes, people can pass through. Thus, the graph can be used for human navigation.

3.2.4 The Safety of the Navigation Route

We show by following theorems that the selected navigation route provides *global safety* as well as *local safety* in the following two theorems. The detailed proofs for those two theorems can be found in Appendix C, available in the online supplemental material.

Theorem 1. *The selected navigation route on the road backbone maximizes the minimum distance of all possible routes to the dangerous areas.*

Theorem 1 shows that global safety, in which the selected route provides user guaranteed safety in global span. Each user along the selected route never moves unnecessarily close to the dangerous areas. The safety is globally guaranteed in the sense that we cannot find another route more distant away from the dangerous areas.

Only providing the global safety, however, sometimes is not sufficient. We have the local safety as the following theorem.

Theorem 2. *For the selected navigation route on the road backbone, any substitute shorter path on the road backbone will not be farther to the dangerous areas.*

Theorem 2 guarantees the local safety, in which that any intermediate local path segment on the selected route yields the largest distance to the dangerous areas. We call this property local safety. Local safety provides the user even stronger safety guarantee at each intermediate step such that at any local step, the selected route guides the user through the safest way.

3.3 Reacting to Emergency Dynamics

Due to the emergency dynamics, the dangerous areas might vary during the navigation process. For example, as a fire spreads, the boundaries of dangerous areas vary from time to time. There are several basic types of variation of dangerous areas, including emerging, expanding, shrinking, and diminishing. For example, the expanding of a dangerous area corresponds to the case that a point beside the dangerous area is switched into the dangerous area. The shrinking of a dangerous area corresponds to the case that a point on the boundary of the dangerous area is switched out of the dangerous area. Obviously, the dynamics introduce problems for the navigation. For example, the original medial axis might no longer provide a safe route under the expanding of dangerous areas. We need to rebuild the road backbone according to the variations of dangerous areas. A straightforward but highly inefficient mechanism is to entirely reconstruct the new

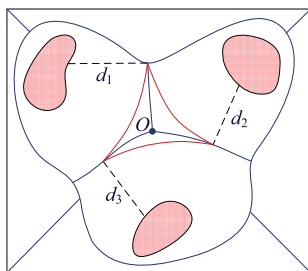


Fig. 4. Improving the road backbone.

road backbone whenever the variation of dangerous areas is detected. Such a mechanism introduces both expensive computation and communication costs to the resource limited sensor network. Furthermore, global collaboration will take relatively longer reaction time, not feasible under the emergent situations.

In this section, we describe an updating principle that incrementally rebuilds the road map according to the dynamics and affects only a local area. In our approach, we let each point in the field maintain a status recording the set of the closest dangerous point and the distance between them. Each time a point is switched into or out of a dangerous area, we only need to update those points that will take it as their closest dangerous point.

We prove in Appendix D, available in the online supplemental material, that any dynamic affects only a local area within the cell of the dynamic dangerous area. All points outside this area maintain their original status.

We further quantify the influence of the dynamics of dangerous areas. According to Appendix E, available in the online supplemental material, in each updating process when the dangerous area varies, our method rebuilds the new road map with the cost of updating a local district of an amortized size $O(\sqrt{n})$. After rebuilding the road map, we accordingly reassign the directions along the road backbone, which incurs local overhead on the compact backbone.

3.4 Improving Route Efficiency

We show how to improve the route efficiency, i.e., providing short routes so as to lead users out of the emergent field rapidly. Unfortunately, the absolute safety of the route by nature contradicts the route efficiency. Achieving the maximum safety leads to the choices of routes “far away” from the dangerous areas, which usually degrade the route efficiency. As Fig. 4 depicts, to maximize the route safety, users have to make a detour across point O . To reduce those long routes, we modify the road backbone, making a trade off between the absolute path safety and the route efficiency.

We find on the road segment between each pair of neighboring cells a point that has the least distance to the dangerous areas in both cells. We name such points inscribed points. Then, in each cell, we build a shortcut between each pair of neighboring inscribed points. The shortcut is a shortest path connecting the two inscribed points that is no closer to the dangerous area than the closest end. Such a path can be built through a constraint flooding within the area of points farther to the dangerous area than the two inscribed points. Fig. 4 depicts the three supplementary shortcuts around the road junction at O .

They are built between the inscribed points on the cell borders, with distance d_1 , d_2 , and d_3 to the dangerous areas. We assign a direction on the shortcut if there is a directional path between its two ends on the original road map backbone and the assigned direction is the same with the original direction. For those shortcuts without directional paths between their two ends on the original road map backbone, we flood the direction assignment information from the two inscribed points.

The building process of the shortcuts only incurs traffic in the local cell between two inscribed points with distance larger than the inscribed points. The area with distance to the dangerous area larger than that of the inscribed points will not be affected. Meanwhile, the areas influenced by building different shortcuts do not overlap. Therefore, the total involved traffic overhead for building all shortcuts is even much less than flooding a packet in the network.

The shortcuts largely improve the route efficiency, reducing long routes on the road backbone. We can treat the route efficiency as a tradeoff by sacrificing the absolute safety in the route. Nevertheless, we show in Appendix F, available in the online supplemental material, that the shortcuts still maintain the global safety.

When there are two or more exits in the field, we can further improve the route safety and efficiency by navigating the users to relatively safer and closer exit. In this case, each exit connects it to the road backbone and floods their gateway information along the backbone. Each point on the backbone selects its direction heading toward the route with the minimum distance to the dangerous areas. Among the routes of the same distance to the dangerous areas, it chooses the shortest one. The resulted directional road is multiple trees rooted at different exit gateways. Again, we can supplement the road map with shortcuts to improve the route efficiency.

4 IMPLEMENTATION EXPERIENCE

To implement such a protocol in practice, we need to carefully address some technical issues when applying our principles in a sensor network.

4.1 Protocol Implementation

In our implementation, we approximate the distance of two nodes by the number of hops along their shortest communicational path. We describe the detailed variables each node maintains and corresponding operations on each node in Appendix G, available in the online supplemental material.

In practice, the density of the network, the sensing range of sensor nodes, and the communication range of sensor nodes indeed have an impact on the safety and efficiency of our proposed approach. Such an impact should be considered in the practical system design. Normally, the dangerous area has a large impact area, for example, the area near a fire often has a relative high temperature. Thus, the dangerous area can be sensed by sensor nodes outside. There are different approaches such as [14], [15] to locate the dangerous area even when there are very limited sensor nodes with limited sensing capability and no sensor nodes are deployed in the dangerous area.

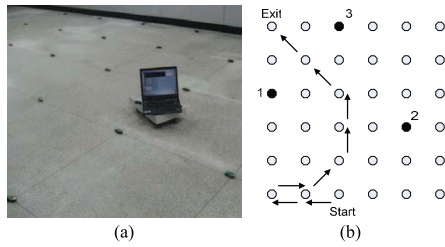


Fig. 5. Experiment setup. (a) The miniature prototype deployment. (b) A showcase of the interactive experiment.

4.2 Prototype Experiment

We implement the navigation system on the TelosB motes and a user interface on the laptop. We describe the detailed system architecture in Appendix H, available in the online supplemental material. We implement a prototype system including 36 TelosB motes deployed into 6×6 grids in the atrium in the university campus, with 10 meter space in-between neighboring nodes. The system provides navigation for a person carrying a laptop computer or PDA that can talk with sensors. Fig. 5a exhibits a miniature deployment of our prototype system in the laboratory [16]. Our navigation protocol builds the road map infrastructure across the monitoring field and provides the route to guide the user out safely. The number of exit is set to 1 in the experiment.

We design a black box challenging game to validate this system. At each step, an internal user is provided the direction pointing from his current stop toward his next stop. The route is represented by a sequence of sensor nodes and the user is directed along those sensor nodes. To decide the direction to move, there are many different ways. For instance, sensor nodes can be equipped with LED lights to indicate the moving direction for the user. Direction antenna can serve as another way to get the moving direction. The direction can be measured by an antenna or sensor array. There are some practical systems, such as [17], based on this technique. Given the distance between the antennas or sensors, those approaches can calculate the direction by measuring the distance to the signal source and the arrival time difference at different antennas or sensors. In addition, we consider the direction estimation error control in our experiment. After the user reaches one sensor node on the constructed road map, the previous direction estimation result will not be further used and the user will launch a new estimation for the next-hop sensor node. It is clear that during the movement, the measurement result is

calibrated in per-hop manner and the estimation error will not accumulate.

At present, we do not equip the laptop with the antenna and sensor array. Alternatively, we configure sensor nodes with the relative direction of each neighbor in advance to facilitate the calculation of directions. Nevertheless, such information will not be revealed to the user so that the experiment truly demonstrates the effectiveness of our system when used in the location-free environment. The other participant uses a PC connecting to the sensor network. He behaves as a challenger to this navigation system, who manages the dangers within the field by setting certain areas from safe to dangerous and vice versa, simulating the emergency dynamics including danger emergence, disappearance, expanding, and shrinking. The frequency and intensity of the update on the dangers represent the extent of the emergency dynamics. Neither of the two participants is aware of the other's operations. The person in the field moves according to the indications received from the navigation system. The challenging person freely sets the dangers without knowing the navigation progress. Such an interactive experiment achieves more than 95 percent success rate and validates the effectiveness of our navigation system under different emergent situations. The 5 percent that are not successfully navigated to the exit is because that the dynamic of the dangerous areas have blocked all possible paths to the exit.

Fig. 5b showcases an instance of the interactive experiment, where the challenger sequentially sets the numbered sensor nodes to be dangerous areas when the internal user moves. Our navigation system accordingly guides the user along the route marked by arrows. Note that the user sometimes needs to go backward to explore safer routes when emergency varies. We show detailed user interface for our experiment in Appendix J, available in the online supplemental material.

Fig. 6a depicts the time to navigate the internal user out of the field. We vary the walking speed of the user as well as the frequency of updating the dangerous areas. The x -axis represents different walking/running speeds of the user and the y -axis represents the time of navigation. Different curves are recorded when we change the dangerous areas with different time intervals. We run 20 tests for each set of parameters. Apparently, a larger walking speed and a lower updating frequency lead to faster navigation. When the walking speed is 3 m/s, our approach reaches nearly optimal navigation time if the environment is

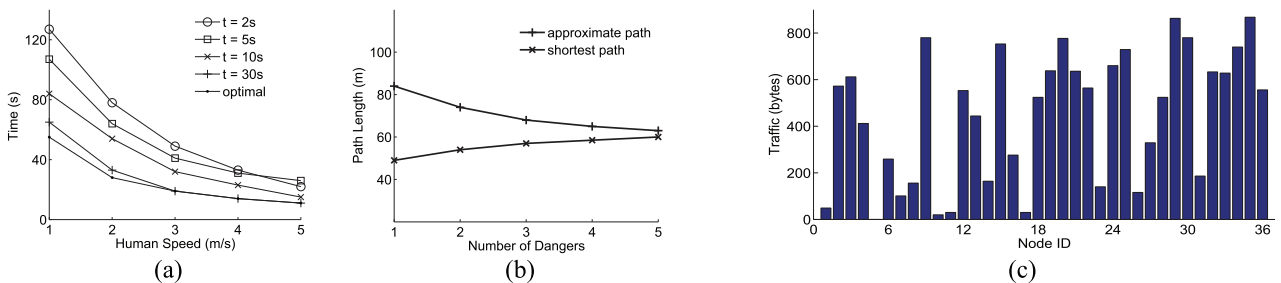


Fig. 6. The experimental results. (a) The time of navigation with different walking speeds and updating frequencies of dangerous areas. (b) The length of the navigation route compared with the shortest path. (c) The traffic cost of each node.

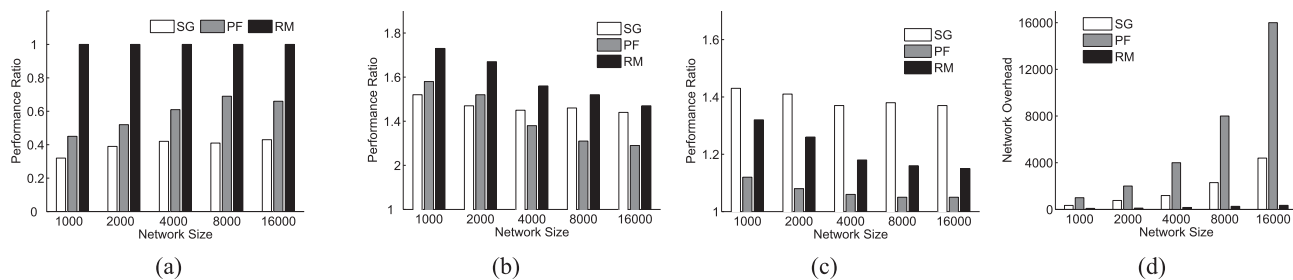


Fig. 7. Comparative results of the three approaches. SG represents the skeleton graph-based approach proposed in [18]. PF represents the potential field-based approach proposed in [19]. RM represents our road map-based approach. (a) Performance ratio of the minimum distance to the danger. (b) Performance ratio to the shortest path. (c) Performance ratio to the minimum exposure path. (d) Average network overhead for updating the network in the event of changes in dangerous areas.

relatively static (30 s updating interval). It reaches less than three times the optimal value even when the environment is highly instable (2 s updating interval for the dangers). Fig. 6b compares the length of the approximate path yielded in our navigation protocol with that of the theoretically shortest safe path, i.e., the shortest path does not pass any dangerous area. As the number of dangerous areas increases, the length of the shortest path increases from less than 50 meters to nearly 60 meters. Contradictive to the intuition, the length of the approximate path provided by our protocol decreases as the number of dangerous areas increases. That is because more dangerous areas will restrict the safest path from going a far way. The ratio of the two lengths decreases from around 2 to nearly 1. Fig. 6c shows the traffic cost in the network. Different nodes have different traffic cost and the total cost is acceptable.

5 PERFORMANCE EVALUATION

We conduct simulations to further evaluate the effectiveness and scalability of our approach. We compare the performance of this design with the skeleton graph based-approach proposed by Buragohain et al. [18] as well as the potential field based approach proposed by Li et al. [19]. We simulate randomly deploying sensor nodes in a rectangular area with an average node degree of 28, the same with that assumed in [18]. To examine the scalability, we vary the size of the field and the number of deployed nodes, while retaining the same network density. The network size ranges from 1,000 to 16,000. For each trial of the network size, we randomly generate 10 internal users in the field to navigate them outward and we take 20 runs, randomly inserting dangerous areas into the field. The number of inserted dangerous areas is uniformly randomly chosen from 3 to 6 and the size of each dangerous area is kept below 5 percent of the total field size. In the evaluation, the number of exit is set to 1. Indeed, our navigation protocol does not rely on any location information before it works. Nevertheless, since the two approaches we compare with all assume the availability of locations, in the simulations, we record the locations of all the nodes and reveal them to those two approaches to facilitate their operation.

For the skeleton graph-based protocol (SG for short), we choose the version based on adaptive skeleton graph, which has been shown superior to the uniform one in their original paper. For the potential field based protocol (PF for short),

we choose the function of calculating the potential value to be $1/dist^2$, which has been used all through in their original paper. We compare the performance of our road map based approach (RM for short) with the two approaches in the following six aspects.

5.1 Minimum Distance to the Danger

We first evaluate the absolute safety of the routes planned in the three approaches. We conduct our tests under static environment with fixed dangerous areas. Let d denote the minimum distance from the planned route to the dangerous areas, and d_{OPT} denote the minimum distance to the dangerous areas from the optimal path that maximizes d . The performance ratio is defined to be d/d_{OPT} . A larger ratio indicates a better safety of the planned route, as the minimum distance from the route to the dangerous areas is larger.

Fig. 7a shows the performance ratio of the three approaches. We can see that the proposed RM approach achieves the optimal result with the ratio = 1. Indeed, such a result is theoretically guaranteed by Theorem 3.10. For the other two approaches, PF has a performance ratio around 0.6 while SG has a performance ratio below 0.4, due to the fact that SG is prone to guide the user close to the dangerous areas to achieve the shortest path on the skeleton graph.

5.2 Shortest Path

We evaluate the path efficiency by comparing the length of the route planned in each approach l with the length of the shortest path that does not cross the dangerous areas l_{OPT} . The performance ratio is defined to be l/l_{OPT} . A smaller ratio indicates a more efficient route, as the route is closer to the theoretically shortest safe path.

Fig. 7b shows the performance ratio of the three approaches under different network sizes. While SG keeps the ratio unchanged around 1.5, PF and our RM have decreased ratio as the network size increases. That is because the hop count-based distance measurement in the two approaches becomes more accurate when more sensor nodes are involved as the network scale increases. When the network size is increased to 16,000, PF reaches the ratio of less than 1.3 and our RM reaches the ratio around 1.5.

5.3 Minimum Exposure Path

By comparing the exposure value of the route planned in each approach with that of the minimum exposure path, we

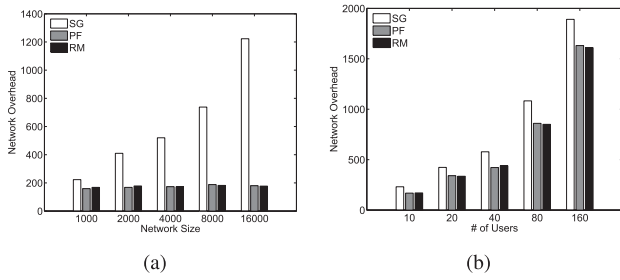


Fig. 8. Traffic overhead for multiple users in the network. (a) Number of messages with respect to network size. (b) Traffic overhead with respect to the number of users in the network.

evaluate the cumulative safety along the route planned in each of the three approaches. The exposure value at each point of the route is calculated as $1/dist^2$. The performance ratio is defined as S/S_{OPT} , where S is the exposure along the route planned in each approach and S_{OPT} is the exposure along the optimal path. A smaller ratio indicates a higher cumulative safety along the path.

Fig. 7c shows the performance ratio of the three approaches under different network size. SG has the highest ratio around 1.4 thus the lowest cumulative safety. PF has the lowest ratio around 1.1. Our RM performs in between with the performance ratio decreasing from 1.32 to 1.15 as the network size increases from 1,000 to 16,000.

Note that such a set of tests indeed favors PF since the function for calculating the exposure value is chosen the same as the function for calculating the potential value of PF maximizing its performance.

5.4 Update Overhead

We evaluate the network overhead incurred by the three approaches for dynamics of the dangerous areas and multiple-user case. We simulate the dynamics of dangerous areas and measure the average message transmissions in each round of network update process. Fig. 7d compares the network overhead of the three approaches. PF incurs the largest overhead as it relies on flooding the entire network to recalculate the potential value of each sensor node and accordingly rebuild the routes. Such a network cost is proportional to the network size. SG introduces relatively smaller overhead, yet proportional to the network size, as in SG the skeleton graph needs to be rebuilt once a dangerous area changes. The proposed RM incurs the least overhead as we have shown in RM only local communication is needed to update the road backbone when dangerous areas change. Hence, RM is scalable as the update overhead is merely proportional to the size of the network.

We also evaluate the traffic overhead for multiple user case. Fig. 8a depicts the number of generated messages within the network. It can be seen that both PF and RM scale well with the network size, generating a small number of control messages. The number of messages involved in SG, however, increases rapidly as the network size becomes larger. This is mainly because in SG different users need to find different paths to the streets in the skeleton graph, resulting in heavy traffic overhead across the network. On the other hand, with PF or RM, once the potential field or the road map backbone has been built, the users only need local information to determine the next move toward to exit. Fig. 8b depicts the traffic overhead with respect to the

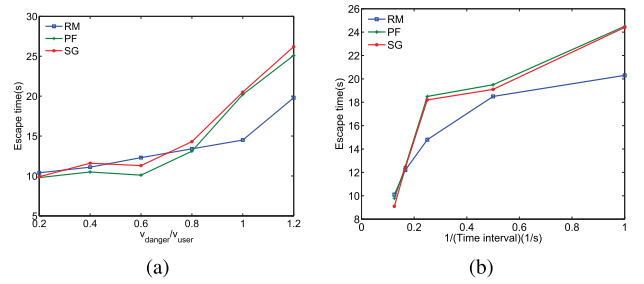


Fig. 9. User escape time with different protocols. (a) The escape time with respect to the v_{danger}/v_{user} ratio. (b) The escape time with respect to time interval of emerging dangers.

number of internal users. The PF and RM approaches only need information of local neighborhood to find out the node as the next stop, so the increase of mobile users introduce minor traffic overhead. On the other hand, the SG approach, which at the beginning has to find out a path to the streets for each user in the skeleton graph, the message cost will increase faster than RM as the number of users increases. We also evaluate average path length for different network sizes and different number of users in Appendix I, available in the online supplemental material.

5.5 Reaction to Environment Dynamics

We simulate the navigation process under various environment dynamics. The simulation is conducted in a 100×100 square area where sensors are scattered in perturbed grids. The communication range of each sensor is 1.5. We vary the danger expanding speed, user escape speed, and the emerging time interval, i.e., how often a danger will emerge, of dangerous areas. The escape time is then recorded under different circumstances. The environment dynamics is totally independent with the user statuses such as their positions and speeds.

Fig. 9a depicts the escape time of users with different protocols. We vary the ratio of v_{danger}/v_{user} , where v_{danger} is the expanding speed of the dangerous area and v_{user} is the speed of user. A larger v_{danger}/v_{user} ratio represents a more dynamic environment. We fix the user speed v_{user} at 5 and vary the expanding speed of the danger. The final result is shown in Fig. 9a. We can see from the results that as the increase of the expanding speed of dangers, the escape time with both PF and SG increases rapidly. The two approaches greedily find the paths which are of minimum hops or minimum exposure values, without taking into account the danger variance. Such paths tend to frequent alternations as the dangerous areas grow. Therefore, the mobile users are more likely to frequently change their navigation paths, leading to a longer escape time. On the other hand, our RM approach, which takes into account the global safety, is able to find those paths that are most likely to be safe in the foreseeable future. Therefore, the path provided in RM is very unlikely to change due to the global safety. Thus, our RM approach lowers the path changing caused by danger expanding, leading to smaller escape time compared with the other two approaches.

In the second simulation, we change the danger emerging interval to examine the escape time for users. More specifically, we keep adding dangerous areas into the network with different time intervals. This aims to simulate the scenario where various danger areas emerge, troubling

existing user navigation paths. We examine the three approaches with different settings of time intervals. Fig. 9b depicts the escape time with different approaches. We can find that as the time interval decreases the escape time of our RM increases much slower than that of PF and SG. This observation verifies again that our road map-based approach that chooses the globally safe path is more adaptive to environment dynamics since the chosen paths are less likely to alter.

6 RELATED WORK

Path planning and navigation are important issues in the fields of robotics [20], [21] and computational geometry [22]. In robotics, the proposed solutions such as [23], [24] try to minimize the path length for the robot to travel from origin to destination while avoiding any obstacles.

For navigation in robotics, based on the information used by robots, we mainly divide existing works into two categories. The first category of methods in robotics navigation is based on the location or geometric information such as works in MOP [25]. More details about this category can be found in Appendix K, available in the online supplemental material, [7], [8], [9], [10], [11]. An essential difference between our work for sensor network navigation and those works in robotics is that we do not assume any location information. We do not even require users' or robots' intelligence to find the path. The navigation is guided by the sensor network system deployed in the field as a supporting infrastructure. Another category of works further enable the robot with perception capability. Robots in those works [26], [27] can perform simultaneous localization and mapping to perceive the physical world. In those works, robots are assumed to be able to sense the environment for path planning, distinguish different locations, and so on. However, using WSNs, the environment can be sensed by cooperatively organized sensor nodes. Users in our system do not need to be aware of extra information outside their surrounding environments. Different from robotics navigation, the self-organized sensor network can provide rich information for navigation. Thus, a key difference between our work and robotics navigation is that in our work the road map is built and maintained by the network of sensor nodes in a distributed manner. Users are not required to compute the optimal path or be aware of the physical world.

In path planning, the most representative example is the flow complex [12]. However, if we want to apply the flow complex to the navigation system, a central controller is needed to collect information from the entire network and further conduct intensive computations, such as differentiation. In addition, once the dangerous area varies (it is a common phenomenon in practice when the dangerous events evolves), in the flow complex, the central controller needs to repeat above information collection and calculation procedures frequently. It is clear that such a method suffers a high communication overhead and a long delay performance in dynamic networks. Different from the flow complex, our approach is a distributed large-scale navigation systems without central controller. Our approach also incurs less computation overhead in the network. Another relevant technique related to our method in path planning is the

maximum clearance path [28]. However, our solution is designed with substantial differences and focuses compared with prior literatures. First, efforts made in previous literatures focus on how to design an efficient centralized algorithm to calculate the maximum clearance path. As a result, those proposed algorithms are suitable for the system deployment in a small region merely. Second, the maximum clearance path pays little attention to the moving direction for robots, since such a management is presumed simple given the global information on hand. Differently, our method introduced in this section is completely distributed and it can assign the moving direction on the road map in an efficient manner.

In sensor network studies, Li et al. [19] first propose a distributed algorithm that explores the minimum exposure path for guiding navigation. Their potential field-based approach largely relies on exhaustive search over the entire network. Buragohain et al. [18] propose to abstract the field by the skeleton graph and accordingly find navigation routes over the skeleton graph. Some studies address the problem of finding the minimum or maximum exposure path in a network. Meguerdichian et al. [29] and Veltri et al. [30] propose heuristics to distributedly compute such paths. Although being similar with our navigation problem, finding the exposure path does not explicitly address the issue of navigating users among dangerous areas, treating individual sensor nodes as adversaries rather than utilizing them as infrastructures. Most of existing studies assume the availability of location information and consider a static field without changes in dangerous areas. Our approach is able to cope with the dynamics of the dangerous area. In addition, the distributed nature significantly reduces the network overhead.

7 CONCLUSION

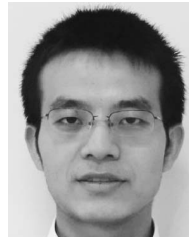
We propose a road map-based approach that provides human navigation in the distributed sensor networks. Primarily different from existing works, we validate our design without relying on location information, which surprisingly overcomes natural intuitions. We further discuss the situation in the event of emergency dynamics, which has not yet been explored by previous studies. We also introduce an updating scheme that locally updates the road map system in the network when the dangerous areas vary, which largely reduces the network overhead. We implement a prototype system consisting of 36 sensor nodes. Through a black box challenging game, we validate the effectiveness of our design. We further evaluate the performance of our approach through large-scale simulations as well as compare it with two existing approaches. The simulation results show that although with much relaxed assumptions, our approach achieves comparable performance with significantly reduced communication overhead.

ACKNOWLEDGMENTS

This study was supported by the NSF China Major Program 61190110, NSFC under grants 61171067, 61202359 and 60970123, NTU SUG M4080103.020, and NAP M4080738.020. The preliminary version of this paper was presented in IEEE INFOCOM 2009 [31].

REFERENCES

- [1] T. He, S. Krishnamurthy, J.A. Stankovic, T. Abdelzaher, L. Luo, T. Yan, L. Gu, J. Hui, and B. Krogh, "Energy-Efficient Surveillance Systems Using Wireless Sensor Networks," *Proc. ACM Int'l Conf. Embedded Networked Sensor Systems (SenSys)*, 2004.
- [2] L. Selavo, A. Wood, Q. Cao, T. Sookoor, H. Liu, A. Srinivasan, Y. Wu, W. Kang, J. Stankovic, D. Young, and J. Porter, "Luster: Wireless Sensor Network for Environmental Research," *Proc. ACM Int'l Conf. Embedded Networked Sensor Systems (SenSys)*, 2007.
- [3] G. Tolle, J. Polastre, R. Szewczyk, N. Turner, K. Tu, S. Burgess, D. Gay, P. Buonadonna, W. Hong, T. Dawson, and D. Culler, "A Macroscopic in the Redwoods," *Proc. ACM Int'l Conf. Embedded Networked Sensor Systems (SenSys)*, 2005.
- [4] C. Wan, S.B. Eisenman, A.T. Campbell, and J. Crowcroft, "Siphon: Overload Traffic Management Using Multi-Radio Virtual Sinks," *Proc. ACM Int'l Conf. Embedded Networked Sensor Systems (SenSys)*, 2005.
- [5] J. Considine, F. Li, G. Kollios, and J. Byers, "Approximate Aggregation Techniques for Sensor Databases," *Proc. IEEE Int'l Conf. Data Eng.*, 2004.
- [6] Z. Li and B. Li, "Loss Inference in Wireless Sensor Networks Based on Data Aggregation," *Proc. ACM/IEEE Int'l Symp. Information Processing in Sensor Networks*, 2004.
- [7] S. Madden, M.J. Franklin, and J.M. Hellerstein, "TAG: A Tiny Aggregation Service for Ad-Hoc Sensor Networks," *Proc. UNIX Symp. Operating Systems Design and Implementation (OSDI)*, 2002.
- [8] D. Guo, Y. He, and P. Yang, "Receiver-Oriented Design of Bloom Filters for Data-Centric Routing," *Computer Networks J.*, vol. 54, pp. 165-174, 2010.
- [9] I. Stojmenovic and X. Lin, "Power Aware Localized Routing in Wireless Networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 12, no. 11, pp. 1122-1133, Nov. 2001.
- [10] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) Using AOA," *Proc. IEEE INFOCOM*, 2003.
- [11] J. Bruck, J. Gao, and A.A. Jiang, "MAP: Medial Axis Based Geometric Routing in Sensor Network," *Proc. ACM MobiCom*, 2005.
- [12] K. Buchin, T.K. Dey, J. Giesen, and M. John, "Recursive Geometry of the Flow Complex and Topology of the Flow Complex Filtration," *Computational Geometry Theory Applications*, vol. 40, pp. 115-137, July 2008.
- [13] B. Kusy, H. Lee, M. Wicke, N. Milosavljevic, and L. Guibas, "Predictive QoS Routing to Mobile Sinks in Wireless Sensor Networks," *Proc. Int'l Conf. Information Processing in Sensor Networks (IPSN)*, 2009.
- [14] Y. Wang, R. Tan, G. King, J. Wang, and X. Tan, "Accuracy-Aware Aquatic Diffusion Process Profiling Using Robotic Sensor Networks," *Proc. Int'l Conf. Information Processing in Sensor Networks (IPSN)*, 2012.
- [15] Y. Zhu, Y. Liu, and L.M. Ni, "Optimizing Event Detection in Low Duty-Cycled Sensor Networks," *Wireless Networks*, vol. 18, pp. 241-255, 2009.
- [16] M. Li, J. Wang, Z. Yang, and J. Dai, "Demo Abstract: Sensor Network Navigation without Locations," *Proc. ACM Conf. Embedded Networked Sensor Systems (SenSys)*, 2008.
- [17] N.B. Priyantha, A. Miu, H. Balakrishnan, and S. Teller, "The Cricket Compass for Context-Aware Mobile Applications," *Proc. ACM MobiCom*, 2001.
- [18] C. Buragohain, D. Agrawal, and S. Suri, "Distributed Navigation Algorithms for Sensor Networks," *Proc. IEEE INFOCOM*, 2006.
- [19] Q. Li and D. Rus, "Navigation Protocols in Sensor Networks," *ACM Trans. Sensor Networks*, vol. 1, pp. 3-35, 2005.
- [20] S. Bhattacharya, N. Atay, G. Alankus, C. Lu, O.B. Bayazit, and G. Roman, "Roadmap Query for Sensor Network Assisted Navigation in Dynamic Environments," *Proc. Second IEEE Int'l Conf. Distributed Computing in Sensor Systems (DCOSS)*, 2006.
- [21] J.C. Latombe, *Robot Motion Planning*. Kluwer, 1991.
- [22] M.d. Berg, O. Schwarzkopf, M.V. Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, second ed. Springer-Verlag, 2000.
- [23] M. Bennewitz et al., "Optimizing Schedules for Prioritized Path Planning of Multi-Robot Systems," *Proc. IEEE Int'l Conf. Robotics and Automation*, 2001.
- [24] F.A. Kolushev and A.A. Bogdanov, "Multi-Agent Optimal Path Planning for Mobile Robots in Environment with Obstacles," *Proc. Third Int'l Andrei Ershov Memorial Conf. Perspectives of System Informatics (PSI)*, 1999.
- [25] E. Frazzoli et al., "Real-Time Motion Planning for Agile Autonomous Vehicles," *Proc. Am. Control Conf.*, 2001.
- [26] Z. Sun and J.H. Reif, "On Robotic Optimal Path Planning in Polygonal Regions with Pseudo-Euclidean Metrics," *IEEE Trans. Systems, Man and Cybernetics*, vol. 37, no. 4, pp. 925-936, Aug. 2007.
- [27] H. Choset and K. Nagatani, "Topological Simultaneous Localization and Mapping (Slam): Toward Exact Localization without Explicit Localization," *IEEE Trans. Robotics and Automation*, vol. 17, no. 2, pp. 125-137, Apr. 2001.
- [28] M. Gavrilova, *Generalized Voronoi Diagram: A Geometry-Based Approach to Computational Intelligence*. Springer Verlag, 2008.
- [29] S. Meguerdichian, F. koushanfar, G. Qu, and M. Potkonjak, "Exposure in Wireless Ad-Hoc Sensor Networks," *Proc. ACM MobiCom*, 2001.
- [30] G. Veltri, Q. Huang, G. Qu, and M. Potkonjak, "Minimal and Maximal Exposure Path Algorithms for Wireless Embedded Sensor Networks," *Proc. ACM First Int'l Conf. Embedded Networked Sensor Systems (SenSys)*, 2003.
- [31] M. Li, Y. Liu, J. Wang, and Z. Yang, "Sensor Network Navigation without Locations," *Proc. IEEE INFOCOM*, 2009.



Jiliang Wang (M'09) received the BE degree in the Department of Computer Science, University of Science and Technology of China, and the PhD degree in the Department of Computer Science and Engineering, Hong Kong University of Science and Technology. His research interest includes wireless sensor networks, network measurement, and pervasive computing. He is a member of the IEEE.



Zhenjiang Li received the BS degree in the Department of Computer Science and Technology, Xi'an Jiaotong University, China, in 2004, and the Mphil degree in the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, in 2009, where he is currently working toward the PhD degree in the Department of Computer Science and Engineering. His research interests include wireless sensor networks and peer-to-peer computing. He is a member of the IEEE.

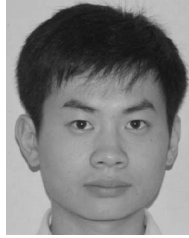


Mo Li (M'06) received the BS degree in the Department of Computer Science and Technology, Tsinghua University, China, and the PhD degree in computer science and engineering from the Hong Kong University of Science and Technology in 2004 and 2010, respectively. He is currently an assistant professor in the School of Computer Engineering, Nanyang Technological University. His research interests include wireless sensor networking, pervasive computing, and peer-to-peer computing. He is a member of the IEEE.



peer-to-peer computing, and pervasive computing. He is a senior member of the IEEE.

Yunhao Liu (SM'06/ACM'06) received the BS degree in automation from Tsinghua University, China, in 1995, the MS and PhD degrees in computer science and engineering from Michigan State University, in 2003 and 2004, respectively. He is currently the EMC Chair Professor at Tsinghua University, as well as a faculty member with the Hong Kong University of Science and Technology. His research interests include wireless sensor network,



and conference, including *IEEE/ACM Transactions on Networking*, *IEEE Transactions on Parallel and Distributed Systems*, *IEEE Transactions on Mobile Computing*, IEEE INFOCOM, IEEE ICDCS, IEEE RTSS, ACM SenSys, etc. He is a member of the IEEE and the ACM.

Zheng Yang (M'06) received the BE degree in computer science from Tsinghua University in 2006 and the PhD degree from the Hong Kong University of Science and Technology (HKUST) in 2010. He is currently a postdoctoral fellow in Tsinghua University and a research assistant in HKUST. His main research interests include wireless ad hoc/sensor networks and pervasive computing. He has published a number of research papers in highly recognized journals

▷ **For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/publications/dlib.**