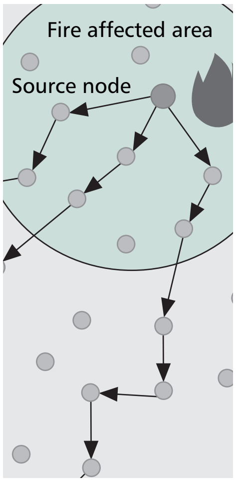
ENABLING MOBILITY IN HETEROGENEOUS WIRELESS SENSOR NETWORKS COOPERATING WITH UAVS FOR MISSION-CRITICAL MANAGEMENT

AYŞEGÜL TÜYSÜZ ERMAN, LODEWIJK VAN HOESEL, AND PAUL HAVINGA,
UNIVERSITY OF TWENTE
JIAN WU, ASML



The authors present a platform that integrates WSNs, UAVs, and actuators into a disaster response setting and provides facilities for event detection, autonomous network repair by UAVs, and quick response by integrated operational forces.

ABSTRACT

Wireless sensor networks have the promise of revolutionizing the capture, processing, and communication of mission-critical data for the use of first operational forces. Their low cost, low power, and size make it feasible to embed them into environment monitoring tags in critical care regions, first responders uniform gear, and data collector sinks attached to unmanned aerial vehicles. The ability to actively change the location of sensors can be used to mitigate some of the traditional problems associated with static sensor networks. On the other hand, sensor mobility brings its own challenges. These include challenges associated with in-network aggregation of sensor data, routing, and activity monitoring of responders. Moreover, all different mobility patterns (e.g., sink mobility, sensor mobility) have their special properties, so that each mobile device class needs its own approach. In this article, we present a platform which benefits from both static and mobile sensors and addresses these challenges. The system integrates WSNs, UAVs, and actuators into a disaster response setting, and provides facilities for event detection, autonomous network repair by UAVs, and quick response by integrated operational forces.

INTRODUCTION

Wireless sensor networks (WSNs) are used to increase the efficiency of many applications, such as target detection and disaster management. The ability and performance of intelligent detection systems providing wireless sensing-computing-actuating are of growing interest. The main objective of the sensing-actuating system is to detect events (e.g., fire) by means of sensors and wirelessly communicate this event and assist other nodes to deliver the event. Generally, sensors deliver their data to specific sink nodes. Sinks collect (sensor) data and act as insertion points of new mission tasks. Most such WSN systems have static sensor and sink nodes that have been developed and also experimentally applied for detection and monitoring activities [1].

However, static WSNs have some limitations. The use of *mobile sensors* could provide significant improvements. They can provide the ability to closely monitor the objects we want to guard in WSNs and look at the events at a smaller granularity than static nodes. Also, *multiple mobile sinks* inside the monitored region can collect the data from the sensors when they pass by. Introducing multiple mobile sinks in WSNs can provide fast energy-efficient data collection with well designed networking protocols.

Mobility of sensor and sink nodes can be achieved by some vehicles or people carrying sensors. It is more efficient to use vehicles instead of people in some cases like disaster management applications due to harsh environmental conditions during a disaster. For this purpose, using aerial and remotely piloted vehicles is a promising idea.

Design and development of a platform that will enable the cooperation of unmanned aerial vehicles (UAVs) with ground wireless sensor-actuator networks comprising static and mobile sensors is the focus of this article. In this article we discuss the requirements, challenges, and opportunities of this platform with a focus on fire detection scenarios.

In what follows we propose a two-layer architecture designed for heterogeneous mobile WSNs as part of European research project AWARE (EU-IST-2006-33579) [2]. Motivated by real disaster management cases, we utilize two system layers of abstraction specialized for specific tasks: dynamic networking under mobility conditions and distributed services. The main concern of this article is dynamic networking under mobility conditions. For this purpose, we focus on the dynamic

 1536-1284/08/$25.00 © 2008 IEEE

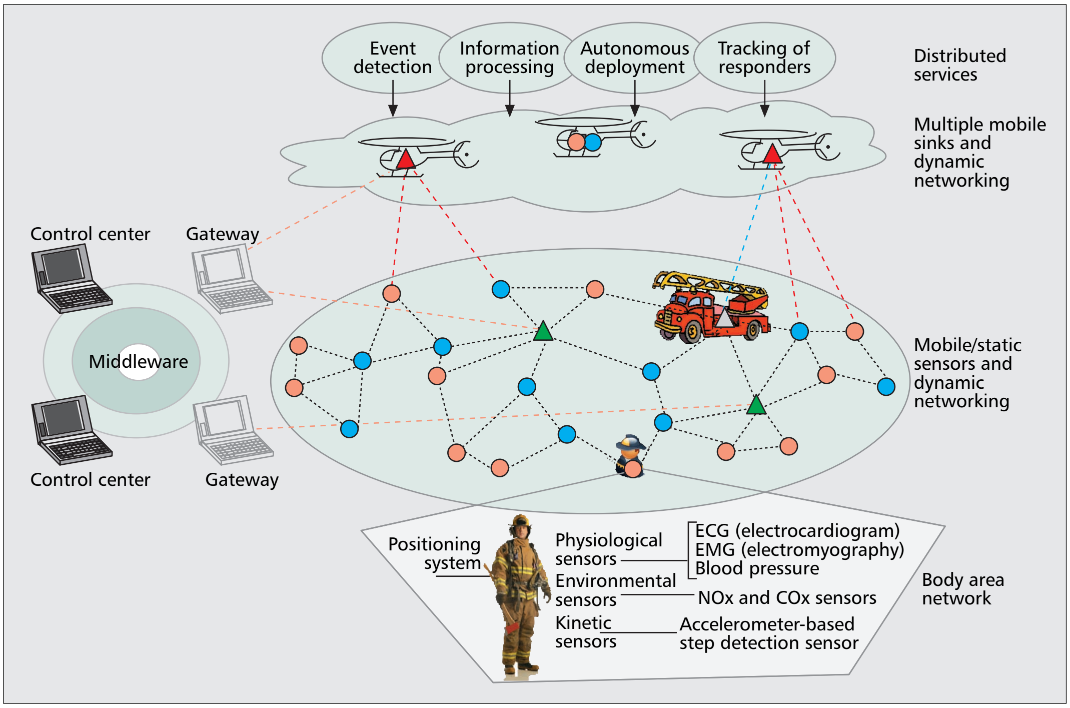


Figure 1. *AWARE wireless sensor network and communication architecture.*

The distributed services layer contains different services to support mission critical management. We have identified four major services with the corresponding opportunities: event detection; information processing; autonomous deployment; and tracking of responders.

aspects and present a new reliable, cost-based, data-centric routing algorithm for such dynamic WSNs. In our research we address, in particular, the success ratio of data delivery with energy cost analysis and the reliability of routing. Instead of each sensor sending its own data report directly to the data sink, we introduce a global-local gradient paradigm to only send aggregated data from the center of the event to the data sink. To increase the reliability of these aggregated data, they are sent via multiple adjustable routes to the data sink. We show the performance of networking under both real mobility in field tests and a simulation environment.

Architecture Overview

The architecture of the AWARE platform comprises a number of heterogeneous subsystems that are described in relation to the global architecture in Fig. 1. We have two key system layers of abstraction: the sensor and dynamic networking layer, and the distributed services layer.

The sensor and networking layer contains the sensor and network protocols that allow messages to be forwarded through multiple sensors taking into account the *mobility* of nodes and the *dynamic change of topology*. In this layer we have multiple mobile sinks attached to vehicles and humans. They can communicate directly with each other via high-speed links and have more processing power. Assignment of each node to a sink in a reliable manner, and handling the dynamics of the mobile sinks and sensors, and change of assignments are the concerns of this layer.

The distributed services layer contains different services to support mission-critical management. We have identified four major services with the corresponding opportunities. *Event detection* supports reliable and timely detection of events. It is even capable of monitoring events in critical regions with mobile sensors. The *information processing* service deals with aspects of collecting and processing data. This service allows vast quantities of data to be easily and reliably accessed, aggregated, manipulated, filtered, disseminated, and used in a customized fashion by applications. *Autonomous deployment* supports detecting routing holes in the network and sends UAVs carrying sensors onboard to these regions to deploy additional nodes. It provides the ability to dynamically adapt the network to the requirements of the situation by increasing the coverage or repairing the connectivity of the network. *Tracking of responders* is also very important for safety-critical events. The *body area network* is used for this purpose. Readings from sensors on responders are collected/processed/integrated to provide better insight into the user's state.

The coordination of the elements in the system is carried out by a control center. The middleware depicted in Fig. 1 provides a publish/subscribe communication interface between all devices such as UAVs, responders, and sensors in the system. Devices that produce data register themselves as data publishers. The middleware then creates the corresponding abstract data channel, which takes care of taking this information to other devices tjat have registered themselves as subscribers to receive the data. Since the middleware tracks the data flow in the AWARE system, it can deliver the statistical data on system functionality to the control center to monitor the state of the system and its components. Also, the collected data can be archived in control centers for future information retrieval.

Both mobile and static sinks, which are typically capable of communicating via multiple interfaces, are deployed on multiple locations. In the case of mobile sinks, mesh networking becomes more complex, because the location or the future location of the sink is not known or predictable.

Network Layer Requirements of Dynamic WSNs in Mission-Critical Applications

In a WSN we can observe generally different data types such as *streams* sent at defined time intervals, *critical data*, and *commands*. In our research we are especially interested in the dissemination of critical data. Important events should be transmitted reliably at all costs. Therefore, our aim is to achieve low latency, high reliability, and a high success ratio of data delivery in routing for mobile WSNs in mission-critical applications:

- *Latency* — The objective of our networking protocols is to minimize latency (i.e., time between the generation of information and the delivery of packet to its destination device).
- *Reliability and success ratio of delivery* — Although wireless transmission and multihop routing can cause packet losses in WSNs, the objective of our networking protocol is to provide reliable communication for critical data. The networking protocol should ensure that the sensing data is being transmitted to the destination successfully in a reliable way.
- *Dynamics and self-adapting* — The networking protocols should be able to deal with mobility of WSN devices. Expected speeds range from running persons to rapidly moving UAVs or vehicles: 0–50 km/h. Moreover, the devices self-organize at power-up and quickly reconfigure as devices join, leave, or move around in the network.

Application Scenario for Field Testing

Protection in natural or human-made disasters is today one of the main concerns of our society. Many countries (e.g., southern European countries) suffer from forest fire devastation every year, with high social, ecological, and economic costs. The application scenario that motivated this work is disaster management, especially fire detection.

To see the functionalities and performance of the system, some field tests were performed where the following WSN with a fire detection setup had been deployed:

- **Multiple (mobile) data sinks** — Both mobile and static sinks, which are typically capable of communicating via multiple interfaces, are deployed at multiple locations. In the case of mobile sinks, mesh networking becomes more complex because the location or future location of the sink is not known or predictable.
- **Static ground sensor nodes** — A set of sensors is deployed in a large geographical area to monitor particular value(s) of interest (e.g., the temperature, humidity, and concentration of toxic materials in the air). UAVs and/or *human operators* typically deploy the ground sensor nodes. The ground network can be fully deployed from the beginning by humans and extended by helicopters if/when needed. Once sensors are deployed, these ground sensors remain at fixed locations and are therefore suitable to act as a reliable communication backbone in the ground WSN.
- **Mobile sensor nodes** — Some time after the fire event is detected, a certain subset of the sensors, collocated in the region, become *hot* in the sense that their readings exceed a predefined tolerance threshold. We would like to ensure the quality of sensor readings for the area bounded by the set of hot sensors. Therefore, *vehicles and UAVs with sensors* are used to collect more reliable data about the event in this dangerous/inaccessible region.

Other mobile sensors are nodes on firefighters. Firefighters carry a set of *physiological*, *kinetic (acting)*, and *environmental* sensors that are depicted in Fig. 1. These sensors form a body area network (BAN), which consists of a set of compact intercommunicating sensors, either wearable or implanted into the human body, that monitor vital body parameters, movements, and environment [3].

Lessons Learned from Field Testing

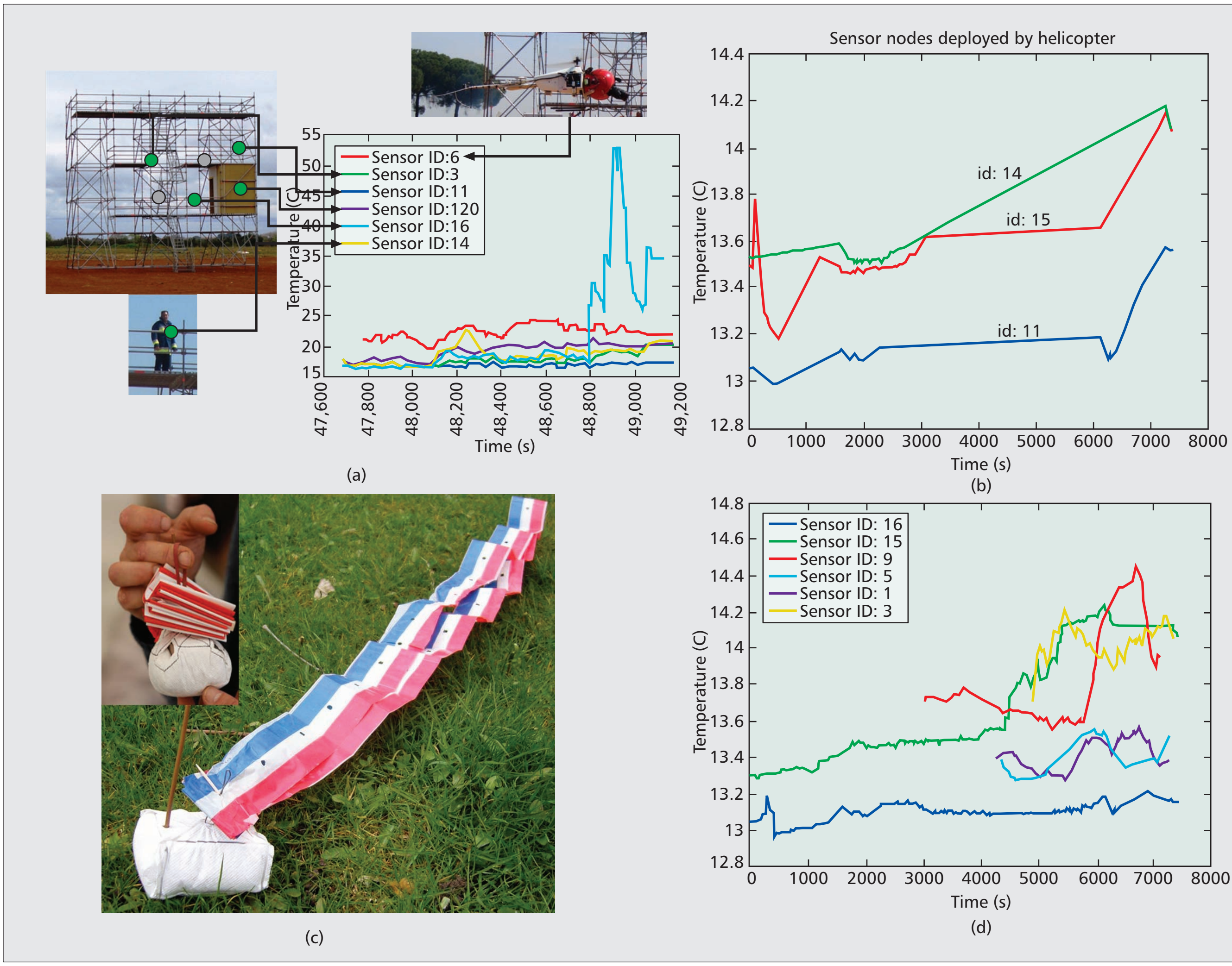
The main objectives of the field tests were to obtain feedback for the design of the platform and record data to develop the different functionalities.

As illustrated in Fig. 2a, a three-floor building was simulated by means of the structure. There is a ladder providing access for firefighters to the three levels. Smoke and fire machines where used to simulate the fires.

The first test was continuous monitoring of environmental conditions, and triggering alarm and local actions when abnormal conditions are monitored. In the temperature diagram of Fig. 2a, the green line corresponds to the data collected from a sensor close to the fire. After the fire is started, the temperature value of sensor 16 rises from 18° C to 54° C and triggers a fire alarm. At the same time, a firefighter carrying sensor 14 starts to walk inside the building. At the sixth time slot (between 48,600–48,800 s), he comes close to the fire and spends a couple of minutes near the fire. However, the increase in the data readings of sensor 14 is low (i.e., 3° C) in this time period. This is due to the small size of the fire and the windy weather conditions. Another important observation obtained from this test is successful data collection from mobile nodes 14 and 6 attached to the firefighter and helicopter. This result shows that the networking functions well under both low mobility (e.g., 4–5 km/h for walking humans) and high mobility (e.g., 40–50 km/h for UAVs).

In Fig. 2b the temperature data collected during the testing of sensor deployment by UAVs is presented. In this test we had sensors 11, 14, and 15 attached to a UAV. While the UAV was flying with sensors on board between 1500 and 3000 s, the sensors' readings were delivered to the sink successfully as shown in Fig. 2b. Deployment was then started, and sensors 11, 14, and 15 were dropped by UAV at times 2250 s, 2625 s, and 3100 s, respectively. In theory, after the deployment, the sink should still be receiving data from the sensors. However, in practice, the sensors lost communication with the sink due to antenna positions after deployment. This result led us to use a special sensor enclosure called a *bean bag* (Fig. 2c), which keeps the antenna perpendicular to the ground, protects the sensor from impact, and prevents jumping.

The aim of the final test was to repair a routing hole by deploying a new sensor using UAVs



■ **Figure 2.** *AWARE experimentation scenarios: a) temperature data collected from WSN; b) sensor deployment by UAVs; c) sensor enclosure bean bag; d) routing hole repairing and data collected from mobile sensor.*

and monitoring the networking during and after deployment. For this purpose, two disconnected clusters were deployed. The first cluster was formed by seven sensors. In Fig. 2d only data collected from nodes 15 and 16 in the first cluster is shown for simplicity. The disconnected cluster consisting of nodes 1, 3, and 5 was also deployed at a far location. As Fig. 2d shows, at the beginning the sink could collect data from only the first cluster (nodes 15 and 16). Self-deployment was then started with a flying UAV carrying sensor 9 at time 3000 s. The sink also established communication with node 9 and started to collect data from this mobile node. At around 4000 s, deployment was done successfully, and nodes 1, 5, and 3 joined the network, respectively. The network could successfully handle the WSN dynamics.

A Routing Protocol for Mixed Mobile/Static WSNs

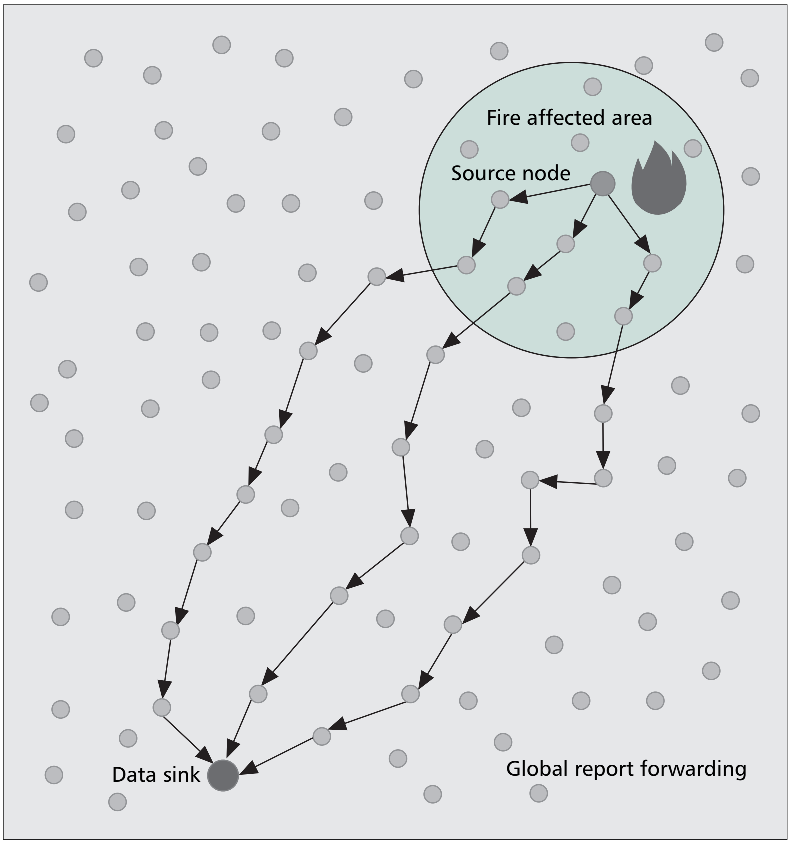
Many data-centric technologies have been proposed to perform in-network aggregation of data to yield energy-efficient dissemination in WSNs [4]. However, current data-centric routing protocols generally assume rather static networks, leading to strong performance degradation in dynamic environments such as gradient broadcast (GRAB) [5].

The basic idea behind GRAB is to make data packets issued by a sensor be delivered along the direction of a sink by decreasing some costs, which are initially built and maintained by the sink but kept by each sensor. The cost at a node is the minimum energy overhead to forward a packet from itself to the sink along a path. The cost field implicitly states the global direction toward the sink. However, GRAB requires each node's cost value to be periodically refreshed by sink initiated flooding, causing the problem of excessive overhead under a dynamic network. Moreover, the mobility of a sink causes network reflooding to reinitiate setup of the cost.

We propose a new reliable cost-based data-centric routing algorithm (RCDR) to deal with the dynamics of WSNs.

Global Gradient and Global Cost Setup

RCDR proposes a *global gradient* paradigm. When a data sink wants to collect data from the network, it sends out a data query to set up a global gradient in the entire network. While this



■ **Figure 3.** *RCDR overview.*

query message propagates in the network, each sensor establishes its own cost value toward this sink. Any data sent toward the sink then follows through the global gradient by multipath routing, as shown in Fig. 3, which increases the reliability of routing. The multipath degree is controlled by the *premium cost* of the data. The *sensor movement adjustment scheme* and *sink movement compensation scheme* efficiently resume the disrupted global gradient by local interactions between sensors. Thus, reflooding, which takes a lot of energy in the network, is reduced to a minimum, while still maintaining the reliability of the network. Moreover, *waiting time* and *forwarding probability* are used to minimize the delay and reduce the broadcast storm problem.

In our WSN each node has a link cost table to all its neighbors. A link cost C_{ij} is the cost between neighboring nodes i and j. When the data sink wants to receive events, the following *global cost setup* steps are executed:

1. The sink sends a data query (DQ) to the network and sets $C_{DQ} = 0$.
2. Each sensor i defines a cost C_i from itself to the sink node, and it initially sets $C_i = \infty$.
3. If intermediate sensor i receives DQ from its neighbor j, it sets $C_i = \min(C_i, C_{DQ} + C_{ij})$.
4. Sensor i sets its lowest cost neighbor (LCN) to sensor j where it could transfer data with the lowest cost.
5. After a random timeout $T_w \in [T_{\min}, T_{\max}]$, sensor i rebroadcasts DQ with $C_{DQ} = C_i$ and a forwarding probability p_f.

The problem with the above method is excessive DQ messages, which prevent it from scaling to a large number of nodes. Before a node settles with minimum cost, it may hear many DQ packets, each of which results in less cost than the previous one. Thus, the node broadcasts many DQ packets. To build the cost field in a scalable manner, we used a similar waiting time approach to that proposed in [6]. Due to space limits, more details of scalability analysis are in [6] which prove that waiting time ensures each node broadcasts only once and with its minimum cost. Therefore, after timeout T_w, any copy of the same DQ message is ignored. After the global gradient setup, each node in the network should have a cost C_i, and the whole network becomes a directed graph toward the sink. The node that did not receive a cost after the global gradient setup (because of collision or errors) will obtain one by the sensor movement adjustment scheme.

Each data unit also has a cost of its own, which consists of two parts. The *basic cost* of data is the cost of the node that sends the data. It can be expressed as $C_{basic} = C_i$. The premium cost of data is set to control the multipath degree of the forwarding paths. When the data has more premium cost to spend, it will travel further away from the low-cost paths and go through higher-cost areas. The premium cost will increase the reliability of forwarding at the expense of more energy consumption. Thus, the overall cost of a data unit is expressed as $C_{data} = C_{basic} + C_{premium}$. When intermediate node j receives new data, it compares the cost of the data C_{data} with its cost C_j. If $C_{data} \geq C_j + C_{ji}$, the data has enough credits to go through this node toward the sink. It will forward the data with a new cost $C_{data} = C_{data} - C_{ji}$. If $C_{data} < C_j + C_{ji}$, the data has no more credits to be forwarded. Node j will drop the data.

Sensor Movement Adjustment

When the data travels in the network toward the sink, it flows through the lower-cost nodes as shown in Fig. 4a. If the connections between nodes ***A*** and ***C*** breaks because node ***C*** moves away from node ***A***, the data from node ***A*** can still go through both nodes ***B*** and ***D***. If the network density is high enough, the data will bypass the troubled link and resume the reliable multipaths as shown in Fig. 4b. If the movement or failure of an individual sensor node breaks the data transfer, a routing hole is formed in the network, and it has to be detected and repaired by a node addition. Networking holes can be repaired by the self-deployment feature of the AWARE system.

The effect of a moved sensor in a new location is also negligible. When a node moves, it could move in two directions in respect to the sink, as shown in Fig. 4c. If node ***A*** moves into a higher-cost area, it will have the lowest cost value among its neighbors. Then it forwards any data from its neighbors, but its neighbors forward none of its data. If node ***A*** moves into a lower-cost area, it will have the highest cost value. Then it forwards no data from its neighbors, but its neighbors forward any of its data. In both cases node ***A*** is excluded from network communications.

On the other hand, if more and more nodes are excluded from the data forwarding, the network becomes very unreliable. A network-wide reset is needed from the sink to restore the gradient field. However, frequently resetting consumes too much energy from the energy restrained sensors. A sensor movement adjustment scheme is designed to minimize the effect of sensor movement by local interactions.

The movement of sensors in the network can be detected by changes of neighbors. A cross-layer approach similar to [7] allows a node to obtain neighbor information from the MAC layer in a very efficient manner. When a new neighbor joins or an old neighbor leaves, the sensor takes different actions to adjust its own cost value.

If node i notices that one of its neighbors j has disappeared, the following rules are executed:

1. If j is LCN, i sets $C_i = \infty$ and sends a cost query (CQ) to its neighbors:
   –Each neighboring node n replies with its own C_n.
   –If sensor i is LCN of sensor n, sensor n also executes step 1 after a delay time.
   –Sensor i sets C_i and LCN according to the steps in global cost setup.
2. If j is not LCN, i ignores the mobility.

If a new neighbor j comes in, the following steps are executed:

1. Sensor j initiates a cost update (CU) message and sets its value $C_{CU} = C_j$ (its cost to the sink).
2. Sensor j sends a CU message to its neighbors.
3. When a neighboring node n receives CU:
   –If $C_n > (C_{CU} + C_{nj})$, it sets $C_n = C_{CU} + C_{nj}$ and LCN to j.

Sink Movement Compensation

Any data loss caused by sink movement decreases the reliability of the network. Particularly in data-centric route scenarios, any movement in the network will disrupt the network setups and result in data losses. When the sink moves, a network-wide broadcast is needed to restore the network gradient. This section introduces a new sink movement compensation scheme with negative gradient, which only requires a local update in order to compensate for sink movement.

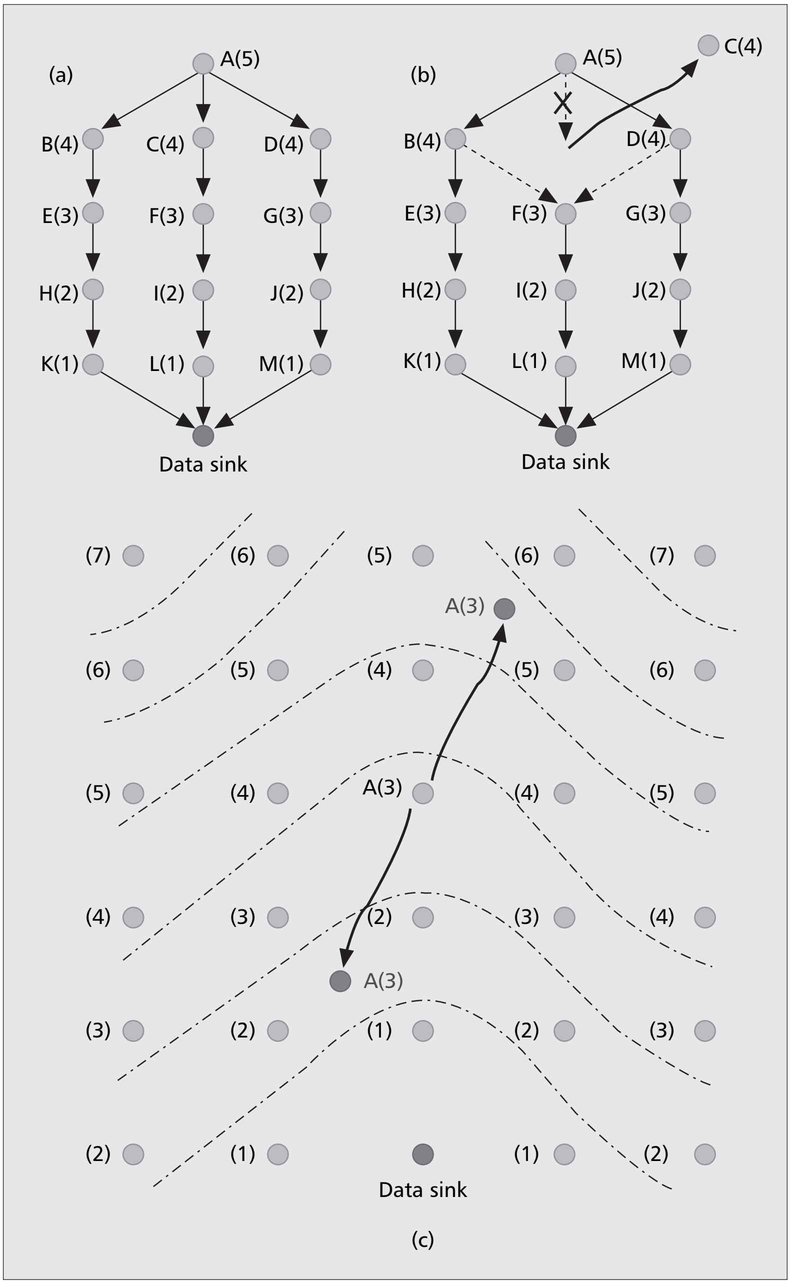
When the sink moves away from its current location, it should be able to first detect its own movement before it can carry out adjustment for the gradients. The movement of the data sink can be detected by an additional localization mechanism. As only a limited number of sinks are needed to collect data, additional hardware on these sinks would not significantly increase the cost of the WSN.

When the sink detects its own movement (or relative movement), it follows the steps below:

1. The sink decreases C_{DQ} to –1 and broadcasts a degrade update (DU) message with a hops-to-live (HTL) field set to h.
2. Each node i receiving DU follows the steps of global cost setup and i sets its new cost as follows:
   –If $HTL > 0$, i lowers C_i and rebroadcasts DU with $HTL = HTL - 1$.
   –If $HTL = 0$, DU is not rebroadcast anymore.

The DU message propagates until it reaches the h-hop neighbors. Thus, all the sensors in this degraded area set their costs one step lower toward the new location of the sink. This creates a small funnel with negative gradient around the sink in the global gradient field. As a result, when the data sent by the node reaches the vicinity of the sink, it will still flow to the new allocated sink by following the small funnel. In this way, a network-wide readjustment is avoided by only a locally restricted gradient broadcast.

The sink repeatedly decreases the cost and increases the HTL of the DU messages when it moves again. So the DU messages can still reach the original location of the sink, the degraded area will expand accordingly. The proposed relationship between the cost and HTL is $h = H - C_{DQ}$, where H is a constant.



■ **Figure 4.** *a, b) Sensor movement in a dense network; c) the effect of sensor movement in two directions.*

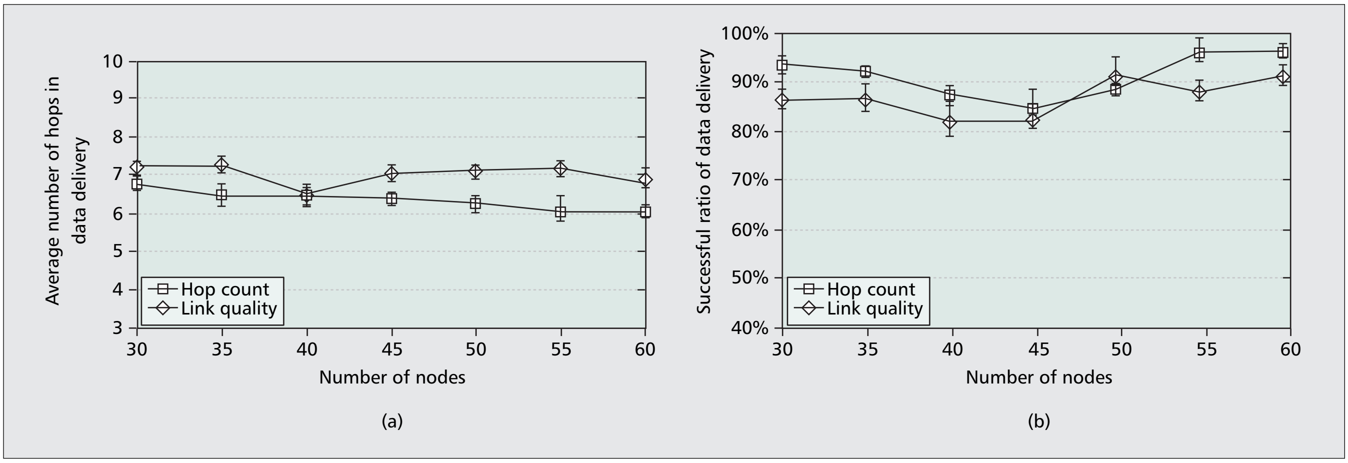


Figure 5. *The average: a) hop counts; b) data delivery success ratio under different metrics of link cost.*

However, the efficiency of routing data through the degraded area decreases as the sink moves further away from its original location. First, the diameter of the degraded area continues to increase. More nodes are involved in the local broadcast when the sink updates the gradient cost. Second, the data sent back by the nodes need to travel more to reach the sink than the shortest possible path. A network-wide gradient reset done by sending out new DQ messages is required to re-establish the gradient field in the network when the diameter of the degraded area gets too large compared to the network diameter.

Simulation Results

We compare our protocol with the GRAB [5] protocol. The same network setup is used to compare the two implementations of routing protocols. The OMNeT++ discrete event simulator, together with a framework for a mobile and wireless network [8], is used in the simulation. For each packet, a link failure probability of 0.02 is applied to the physical layer. For both simulations Sensor-MAC protocol (SMAC) [9], a medium access protocol for wireless sensor networks, is implemented to provide MAC layer access. It is a carrier sense multiple access with collision avoidance (CSMA/CA) protocol. A network of a certain number of sensors with a fixed radio range is randomly placed in a rectangular area. The nodes move in this area according to the random waypoint model (RWP), which leads to worst case analysis of RCDR with random speeds and waiting times.

Link Cost Analysis — The selection of link cost value decides the random forwarding character of the sensor network. For example, a delay related cost would make this WSN have overall lower delay. In the following simulations we try to show the effect of two metrics of the link cost: hop count and link quality.

Each link was assigned a fixed failure probability P between 0 and 0.1. For these two metrics of link cost, we run the simulations under the same topology.

We choose simple relationships between link cost and the metrics to show their effect on the data forwarding:

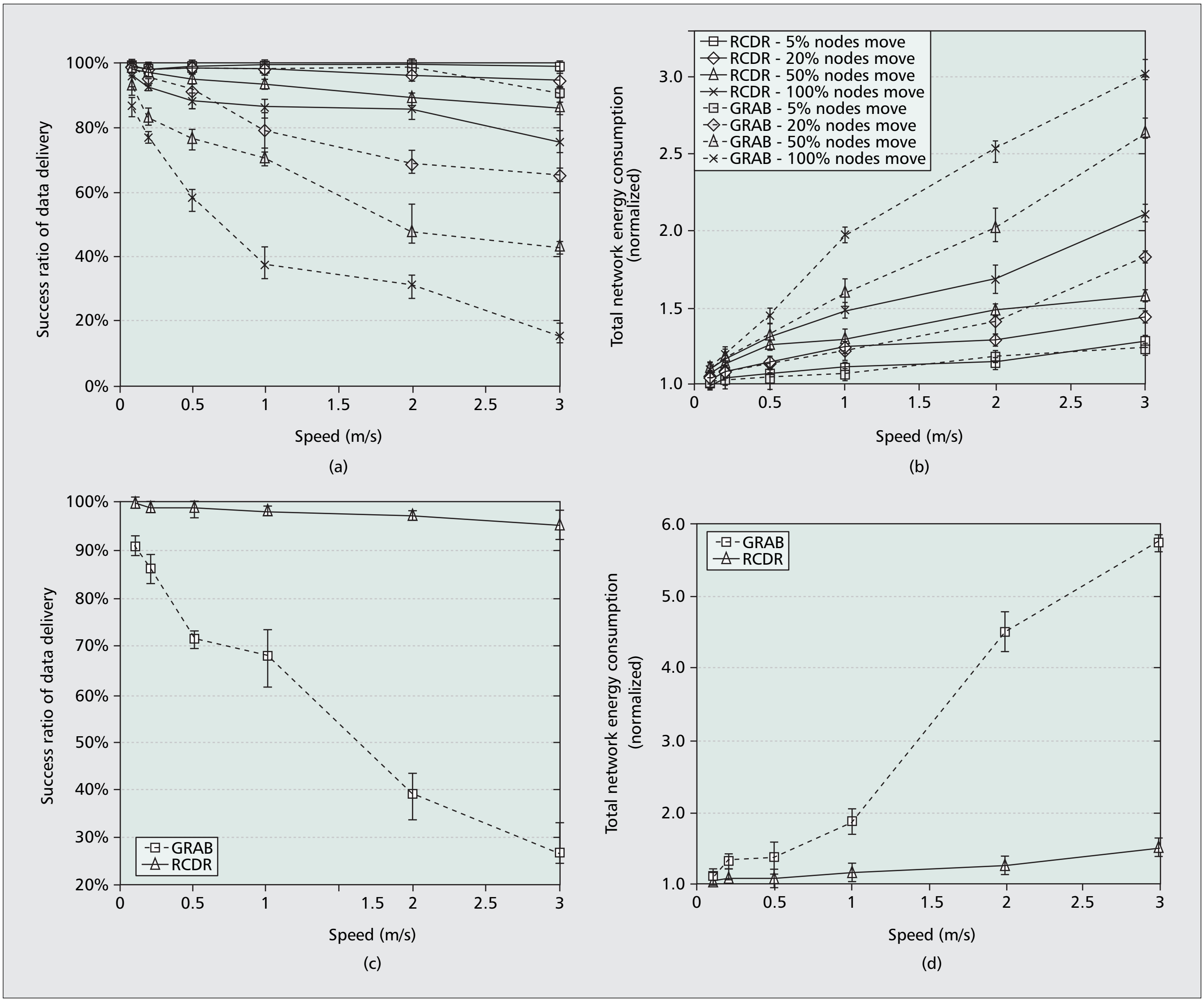
- Hop count: Each link has the same link cost $C_{link} = 1$. The message will be forwarded according to the lowest hop count routes.
- Link quality: The link cost is expressed as $C_{link} = 1 + 10P$. The higher the link quality, the lower the link cost. Therefore, a more reliable link has a higher change rate of forwarding data.

The results in Fig. 5 clearly show that hop-count-based link cost gives the network lower delay than the link quality metric. Another observation is that both hop count and reliability based networks have high success ratios of data delivery.

Mobility Analysis — In the first simulation we try to discover the reliability of the sensor movement adjustment scheme under different mobility conditions. A network of 60 sensors with a radio range of 150 m is randomly placed on a rectangular area of 800 m × 800 m. We only set up a global gradient in the network and let one random sensor in the network generate data reports to the data sink. This gives a data flow for 10 min with a data rate of 2 packets/s, after which another random node takes over. A certain percentage of sensors in the network move in the network, and the other sensors remain static during the course of the simulation.

First, we compared the success ratio of data delivery between RCRD and GRAB under different movement speeds. As shown in Fig. 6a, when only 5 percent of the sensors in the network move at a speed of less than 2 m/s, both RCRD and GRAB are rather reliable. When more sensors move in the network, successful delivery by GRAB decreases sharply; when the speed is more than 2 m/s, its success delivery ratio is less than 50 percent. This means that GRAB is very unreliable and almost nonoperational in a dynamic network. On the contrary, RCDR shows much better resilience to the changing topology due to the sensor movement adjustment scheme.

Second, we compare the energy consumption of the whole network under different speeds. As shown in Fig. 6b, at low speed both protocols have similar energy consumption. When the speed increases, GRAB consumes more than two times the amount of energy as RCDR, which is caused by its more frequent network-wide flooding to resume the gradient. However, the



■ **Figure 6.** *a) The data delivery success ratio; b) the network energy overhead of RCDR and GRAB under different speed of sensor movement; c) the data delivery success ratio; d) the network energy overhead of RCDR and GRAB under different speed of sink movement.*

local adjustment of RCDR shows its advantage at higher speeds in 50 percent savings in energy consumption.

In the final simulation set, we tried to find the reliability of the network under the sink movement compensation scheme. All sensors except the data sink remained static during the course of the simulation. The sink follows the RWP model at different speeds. Figure 6c clearly shows that compared to GRAB, the sink movement compensation scheme improves the reliability of the network by 20 percent at lower speeds and more than 75 percent at higher speeds. At the same time, its energy consumption is only 25–50 percent that of GRAB. The "disaster" situation of sink movement in GRAB is solved well by our scheme.

Conclusions

In this article we address the opportunities and challenges of integration of mobile sensor network technologies into disaster management applications. The ultimate goal is to use the advantages of mobility with low-cost embedded devices and thus improve the response time in mission-critical situations. We present a data-centric routing protocol that supports establishment of a global gradient that only sends the aggregated data from the center of the event to the data sink via multiple adjustable routes to increase the reliability. Also, the global gradient is supported by sensor and sink movement schemes designed to resume the network gradient when the network topology changes. The simulations and field tests confirm the feasibility and reliability of our routing protocol.

Consequently, we consider as future work the study of a quick network hole detection algorithm, which is essential for routing hole repair in self-deployment by UAVs.

Acknowledgments

This work has been sponsored by the European Commission as part of the AWARE project (IST-2006-33579).

REFERENCES

[1] A. Mainwaring *et al.*, "Wireless Sensor Networks for Habitat Monitoring," *WSNA '02*, Atlanta, GA, Sept. 28, 2002.

[2] Platform for Autonomous Self-deploying and Operation of Wireless Sensor-Actuator Networks Cooperating with Unmanned Aerial Vehicles (AWARE); http://www.aware-project.net/

[3] E. Jovanov *et al.*, "A Wireless Body Area Network of Intelligent Motion Sensors for Computer Assisted Physical Rehabilitation," *J. NeuroEng. and Rehabilitation*, vol. 2, no. 6, Mar. 1, 2005.

[4] B. Krishnamachari, D. Estrin, and S. Wicker, "Modeling Data-Centric Routing in Wireless Sensor Networks," *IEEE INFOCOM '02*, vol. 2, New York, NY, 2002, pp. 39–44.

[5] Ye, S. Lu, and L. Zhang, "GRAdient broadcast: A Robust, Long-Live Large Sensor Network," UCLA Comp. Sci. Dept., tech. rep., 2001.

[6] F. Ye *et al.*, "A Scalable Solution to Minimum Cost Forwarding in Large Scale Sensor Networks." *10th Int'l. Conf. Comp., Commun., and Net.*, 2001.

[7] L. van Hoesel *et al.*, "Prolonging the Lifetime of Wireless Sensor Networks by Cross-Layer Interaction," *IEEE Wireless*, Special Issue on Wireless Sensor Networks, vol. 11, no. 6, 2004, pp. 78–86.

[8] S. Dulman and P. Havinga, "A Simulation Template for Wireless Sensor Networks." *Proc. IEEE 6th Int'l. Symp. Autonomous Decentralized Sys.*, Pisa, Italy, 2003.

[9] W. Ye, J. Heidemann, and D. Estrin. "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," *IEEE INFOCOM '02*, vol. 3, New York, NY, June 2002, pp. 1567–76.

BIOGRAPHIES

AYŞEGÜL TÜYSÜZ ERMAN (a.tuysuz@utwente.nl) received a B.Sc. degree in computer engineering from Yeditepe University, Turkey, in 2004 and an M.Sc. degree in computer engineering from Bogaziçi University, Turkey, in 2006. From October 2005 to June 2006 she was an affiliated researcher at the Satellite Networks Research Laboratory, Bogaziçi University. She is currently pursuing her doctoral studies within the Pervasive Systems group at the University of Twente, the Netherlands. She is working within the European project AWARE on mobility aspects of mission-critical wireless sensor networks. Her major research interests are wireless mobile sensor networks and mobility-aware networking management.

LODEWIJK VAN HOESEL (l.f.w.vanhoesel@utwente.nl) holds a Master's degree in electrical engineering and a Ph.D. in wireless sensor networks from the University of Twente. In the recent past he has been working on energy-efficient medium access control. He is currently working within the European project AWARE on energy-efficient wireless sensor networks for safety-critical missions. Research topics include physical layer, medium access protocols, and routing protocols.

JIAN WU (jian.wu@asml.com) studied at Beijing University of Posts and Communications, China, from 1995 to 1999 and received a B.Sc. degree in electrical engineering there at 1999. From the University of Twente he received both his M.Sc. in computer science and Ph.D. in 2002 and 2007, respectively. In November 2006 he joined the R&D department of ASML, working on middleware integration for advanced lithographical systems.

PAUL HAVINGA (p.j.m.havinga@utwente.nl) is an associate professor in the Computer Science Department of the University of Twente, and founder and CTO of Ambient Systems, Enschede, the Netherlands. He received his Ph.D. with a thesis entitled "Mobile Multimedia Systems" in 2000, and was awarded the DOW Dissertation Energy Award for this work. His research themes have focused on wireless sensor networks, ambient intelligence, distributed systems, energy-efficient wireless communication, and mobile computing. The common theme in these areas is on the development of large-scale heterogeneous wireless distributed systems. Research interests cover architectures, protocols, programming paradigms, algorithms, and applications. This research has resulted in over 200 scientific publications in journals and conferences. As founder of a wireless sensor network company, he has received various awards. In May 2007 he received the ICT Innovation Award for the successful transfer of knowledge from university to industrial use. In June 2007 he received the van den Kroonenberg award for being a successful innovative entrepreneur. He is project manager of several large international projects, is involved as program committee chair, member, and reviewer for many conferences and workshops, and regularly serves as an independent expert for reviewing and evaluation of international research projects for the EU, the United States, and international government.