Computers and Electrical Engineering 41 (2015) 407–425

Contents lists available at ScienceDirect

Computers and Electrical Engineering

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

Wireless sensor network-based communication for cooperative simultaneous localization and mapping ☆

Gurkan Tuna [a,*], Vehbi Çağrı Güngör [b,c,e], Stelios M. Potirakis [d]

[a] Department of Computer Programming, Trakya University, Edirne, Turkey
[b] Department of Computer Engineering, Bahçeşehir University, 34353 Besiktas, Istanbul, Turkey
[c] Department of Computer Engineering, Abdullah Gül University, 38039 Kayseri, Turkey
[d] Department of Electronics Engineering, Technological Education Institute of Piraeus, Aigaleo, Greece
[e] Akademi Ar-GeYazilim ve Muhendislik, Erciyes Teknopark A.Ş., 38039 Kayseri, Turkey

ARTICLE INFO

Article history:
Available online 1 April 2014

ABSTRACT

This paper presents a novel approach of using a Wireless Sensor Network (WSN) as the communication means for Multi-Robot, Cooperative, Simultaneous Localization and Mapping (CSLAM) applications investigating the associated design challenges and suggesting corresponding solutions. Although the proposed approach brings several benefits including an increased coverage and communication range, self-organization capabilities, quick deployment, and flexible architecture, the realization is interrelated with performance in terms of energy efficiency and reliability. In this respect, the applicability of the WSNs for the presented approach is investigated. Centralized and distributed map merging methods in WSN-based CSLAM are evaluated in detail and the impacts of packet delays and losses on the performance of CSLAM algorithms are shown. Additionally, the involved network congestion and contention dynamics are presented, while the effects of observation range, speed, time intervals between observations, and odometry readings on the SLAM accuracy are shown based on an extensive set of simulation studies.

1. Introduction

Simultaneous Localization and Mapping (SLAM) is a process used by mobile robots to build a map of an unknown zone by using a sequence of measurements, while at the same time they keep track of their current locations [1]. Due to the measurement noise and motion noise, uncertainty is inherent in robotic mapping [1–3]. A team of multiple robots can map an unknown zone more quickly and robustly than a single robot. This process is known as Cooperative SLAM (CSLAM) [4]. SLAM algorithms rely on the environment representations, maps, which consist of a set of features detectable by the robot sensory system [3]. There are three well-known map representations in SLAM, namely the occupancy-grid maps, the topological maps, and the landmark-based maps. In the case of the occupancy-grid maps, the environment is represented in a discrete grid which is composed of cells. Each cell is assigned a value which represents the probability of occupancy. Grid resolution is the key variable of occupancy-grid maps [3]. As the grid resolution decreases, average CPU-time and memory requirement needed for global localization decreases but average localization error increases. The disadvantages of occupancy-grid maps are that they suffer from discretization errors and require a lot of memory resources. Topological maps

☆ Reviews processed and recommended for publication to Editor-in-Chief by Associate Editor Dr. Sherali Zeadally.

* Corresponding author. Tel.: +90 284 2240283.
E-mail address: gurkantuna@trakya.edu.tr (G. Tuna).

http://dx.doi.org/10.1016/j.compeleceng.2014.03.003
0045-7906/

consist of vertices and edges. A vertex represents a specific place and an edge indicates the traversability between two connected vertices. Landmark-based maps store landmarks' locations and robots' positions in state vectors. In a two-dimensional landmark-based map, the location of a landmark is stored in the Cartesian coordinate system. A covariance matrix associated with the map is used to describe the uncertainties of landmarks' locations and robots' positions [4]. The memory size required to store a landmark-based map is very small comparing to an occupancy-grid map or a 3D map.

This paper presents the design considerations of using a Wireless Sensor Network (WSN) as the communication means for CSLAM and investigates the potential advantages and design challenges. In CSLAM applications, robots come together at some predetermined locations to share their findings. This makes the operation slower. If communication is available all the time, then the operation may run faster. In this respect, the use of WSNs can improve communication between robots and a control center responsible for overall coordination for both centralized map merging and distributed map merging approaches, and can address the main problem of outdoor CSLAM applications, which is the lack of communication between the robots and the control center (or human operator on the semiautonomous mode).

Different from the traditional approaches that use IEEE 802.11b/g/n-based wireless networks to communicate between robots and a central agent, the efficient use of WSNs is proposed here for the distributed communication and coordination in CSLAM. According to this approach the wireless sensor nodes, beyond their main task of sensing the environment to collect information, provide a communication infrastructure to mobile robots. The proposed WSN-based CSLAM has a potential use for robotic exploration and SLAM applications especially in hazardous environments or situations like nuclear disaster zones, chemical attack zones, minefields, even conventional battlefields or natural disaster sites. It is important to note here that, in addition to adapting WSNs to the needs of CSLAM, the proposed SLAM strategy needs to be adapted to the features of the employed WSNs.

This paper extends the conference publication [5] of Tuna, Gulez and Gungor with more details about the proposed system, extensive new results and findings from a set of simulation studies on the performance of the centralized and distributed map merging methods in WSN-based CSLAM. Different from [5] which only address WSN communication performance in terms of both node level parameters and network level parameters for the proposed application, in this paper, the advantages of the proposed WSN-based CSLAM and the associated communication challenges are thoroughly investigated in order to focus on designing strategies to deal with the limitations of WSNs for CSLAM applications. In addition, the problem statement is described in more detail, while all pictures and result graphs have been updated to the new findings of the ongoing work. The main contributions of this paper can be summarized as follows:

- The key WSN design issues and challenges for CSLAM are revealed. In addition, the relationship between data transmission rate and lifetime of contemporary sensor network platforms, such as mica2, imote2, and Telos sensor motes is identified by simulation.
- Centralized and distributed map merging approaches in WSN-based CSLAM are evaluated, and the impacts of packet delays and losses on the performance of SLAM algorithms are shown. The dynamics of network congestion and contention in WSN-based CSLAM are investigated, and the effects of the observation range, the speed, the time interval between observations, and the time interval between odometry readings on the SLAM accuracy are shown based on an extensive set of simulation studies.
- The nonlinear relationship between the number of data transmissions in WSNs and battery voltage is demonstrated through empirical field measurements using Telos sensor network platforms in the field.

The remainder of the paper is organized as follows. A novel WSN-based communication for CSLAM is introduced in Section 2, while energy consumption issues for the proposed approach are investigated in Section 3. In Section 4, the design considerations of WSNs for CSLAM are studied in detail. In Section 5, centralized and distributed map merging methods in WSN-based CSLAM are examined. Finally, the paper is concluded in Section 6.

2. WSN-based communication for CSLAM

In CSLAM applications, robots meet at predetermined locations to share their findings, the map data. This necessity is due to the communication unavailability at some points and is the main limitation of all CSLAM applications. To address this limitation, in this study, we propose the use of WSNs to enable communication between robots.

Map merging is one of the most important steps of CSLAM applications. It is basically the process of building a consistent model of an unknown zone by using sensor data collected from different robots. In the centralized map merging approach, observations from all robots are used to build a map [6]. On the other hand, in the distributed approach, each robot independently builds the local sub-map of the zone around itself. Then, local sub-maps are fused into a global map periodically [4,7,8]. In the proposed approach, since the robots exchange map data through WSN communication channels, the performance of CSLAM map merging operation heavily depends on the performance of WSNs.

In general, WSNs offer many advantages like flexible installation and maintenance, fully mobile operation, and monitoring of environments inhospitable for humans, however they have some adverse properties due to the wireless channels used during transmission such as path loss, multi-path fading, and channel interference. Thus, WSNs are inherently, up to a certain degree, unreliable and unpredictable, which has been the case for many of the sensor network deployments [9,10]. However, recent work on radio diversity shows that WSN links can be nearly perfectly reliable with multiple radio bands [11]. A dual radio network architecture can be used to improve communication reliability in WSNs, which can be further integrated into the proposed approach. Overall, in order to realize the proposed approach successfully, energy efficiency and reliability issues need to be investigated and the novelty of this study lies in design strategies to deal with the limitations of WSNs for CSLAM applications.

Contrary to the traditional approaches which use IEEE 802.11b/g/n-based wireless networks for communication between robots and a central agent, in the proposed approach WSNs serve as a communication infrastructure for distributed communication and coordination in Multi-Robot SLAM as shown in Fig. 1. In this approach, robots are equipped with exteroceptive sensors such as Laser Range Finders (LRFs), Ultrasonic Range Finders (URFs), and Infrared Range Finders (IRFs) to perform CSLAM. The robots also have IEEE 802.15.4 interfaces to communicate over a WSN. In addition to its main task of distributed sensing of a physical phenomenon, the WSN also serves as a communication medium between the robots. The main advantage of this approach is that WSNs provide a distributed communication topology. Wi-Fi also supports an ad hoc mode which supports peer-to-peer communication, and this can also be used for multi-robot systems. However, if Wi-Fi is used instead of WSN for CSLAM it is required to use a large number of robots to reach the same extent of coverage. If the cost of mobile robots and the cost of sensor nodes are compared, the benefit of using the proposed system can easily be seen. Moreover, to extend the lifetime of the deployed system, low power and energy-efficient communication protocols need to be developed. Especially, since packet transmissions affect the battery lifetime significantly, packet losses leading to waste of battery lifetime should be avoided as much as possible.

The performance of WSNs depends on several factors including the average distance between the deployed sensor nodes, the transmitted power, the available battery power, packet size, and the reporting frequency. The following list refers to issues that have to be confronted for WSNs to be appropriate for CSLAM applications.

1. Since different sensors have different capabilities like the available processing power, the available battery power, and antenna features, and all these capabilities affect the performance of a WSN, wireless sensor selection for robots is an important design step for WSN-based CSLAM.
2. Depending on the type of the application and the environment, WSN nodes can be deployed by using different strategies. The distances between the wireless sensor nodes need to match the sensors' specifications, while they determine the robustness characteristics of the WSN.
3. If there are some factors in the environment preventing the communication between sensors, external antennas can be used to improve the communication.
4. Since transmit and receive power affect the WSN performance and the lifetime of sensor nodes, the tuning of transmit and receive power parameters is important.
5. Reporting frequency of wireless sensor nodes affects the WSN packet reception rate and the lifetime of sensors' batteries.
6. Before the real deployment, a test phase can be carried out to examine the possible packet losses, transmission delays and if necessary, new nodes can be added to improve the communication performance of the WSN.

On the other side of the integration, there are SLAM related issues that need to be addressed. The following list refers to important issues that have to be addressed for CSLAM applications to successfully use WSNs.

1. *Coordination problem:* Coordination and cooperation of robots with unknown relative positions among each other are challenging, when they operate cooperatively to build a map.

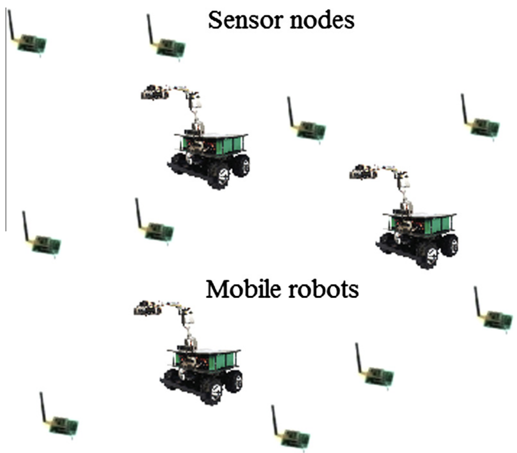

Fig. 1. A WSN-based CSLAM includes mobile robots that are responsible for building a global map, and wireless sensor nodes that together act as the communication infrastructure of the CSLAM application.

2. *Data association issue:* This can be described as "finding correspondences between robot sensor measurements and map landmarks". Since only a single, common, map of the environment, which is shared between all sensors, is maintained it is required to make correspondences between the sensor measurements and the environment landmarks in the map [6].
3. *Limited communication resources:* During exploration of large-scale WSN-based environments, communication between the robots might fail due to the presence of inactive sensors or local deadlocks.
4. *Accuracy vs. processing load trade-off:* Robots learn their positions by using sensors to acquire the surrounding environment information. However, it is not possible to sample the whole environment with all details due to the processing time and processing load restrictions. Hence, some samples are taken from the environment [12].
5. *Central vs. distributed processing:* One needs to design algorithms that specify how/where the computations for SLAM will be handled. If a central controller is employed, how is the communication going to be handled? If the processing is performed in an ad hoc sensor network, how is the exchange of data between the nodes going to be done?
6. *Map updating:* One needs to define appropriate criteria to determine how to update the common map based on the observations from various sensors [13]. In multi-robot applications, map updating depends on the type of the map merging strategy used: Centralized or distributed.

In this study, we specifically investigate network congestion and contention in WSN-based CSLAM applications. Packet delays and losses are typical results of the network congestion and contention. Therefore, we analyze the impacts of the packet delays and losses on the performance of CSLAM applications that are based on both centralized and distributed map merging approaches. In addition, we evaluate the effects of several parameters including observation range, speed, time interval between observations, and time interval between odometry readings on the accuracy of CSLAM.

3. Energy consumption issues in WSN-based CSLAM applications

A wireless sensor node comprises a micro controller unit (MCU), a memory unit, a transceiver, a battery and one or more sensors. Various wireless sensor nodes are available commercially. Since they have integrated batteries, their energy resources are limited. In WSNs, energy efficiency is crucial due to the constrained energy resources of the sensor nodes. Conventional WSNs have been designed for low-data rate applications such as event detection, periodic measurements, and tracking. Hence, an application that requires frequent data transmissions, such as CSLAM, brings several challenges that need to be addressed.

In order to investigate the relationship between data transmissions in WSN-based CSLAM and the lifetime of different types of sensor nodes, a simulation environment using MATSNL [14] was developed. In MATSNL, if a schedule-driven model is employed, the processors of the sensor nodes directly drive the sensors in order to wake the processors up to sample the sensors' output according to a schedule [14]. For this model, detection probability is basically the duty cycle of a node. In WSNs, duty cycle can be described as the ratio between active period and the full active/dormant period of a node [15,16]. If the schedule-driven model is modeled as a Markov chain, the average steady-state power consumption of a node can be formulated using (1) [17].

$$\overline{P}_{SD}(u) = (P_W(\lambda) - P_{S0})u + \left(P_{S0} + \frac{C_P}{T_c}\right) \tag{1}$$

where P_W represents the power consumption at awake period of a node, P_{S0} represents the power at asleep period, C_P represents the wake-up energy cost of the MCU, T_c represents duty period, u represents detection probability, and λ represents the average event inter-arrival rate.

On the other hand, if a trigger-driven model is employed, the sensors coupled to the preprocessors of sensor nodes sense the environment and wake up the rests of the nodes as soon as an event is detected. If the trigger-driven model is modeled as a Markov chain, the average steady-state power consumption of a node can be formulated using (2) [17].

$$\overline{P}_{TD}(\lambda) = \frac{P_{S1} + \lambda K_E}{[1 + \lambda K_T]} \tag{2}$$

Eq. (2)'s power components of can be separated into two parts. λK_E represents the average power spent for computation and communication for each sensed event, P_{S1} represents the power spent to monitor the events. Where K_E represents the energy spent for a sensed event, K_T represents the average time for a sensed event, and λ represents the average event inter-arrival rate. Detailed information on deriving the equations of the models can be found in [17].

Fig. 2(a) and (b) shows the results for different commercially available wireless sensor nodes, such as the "mica2" [18,19], the "imote2" [20], and the "Telos" [19] nodes. For the 1st case study, the duty cycle was set to 1 and event inter-arrival rate ranging from 1 s to 1 day was assumed. For the 2nd case study, the duty cycle ranging from 0.2 to 0.8 was assumed and event inter-arrival rate was set to 1 day. For a fair comparison, each platform's power consumption and its features should be taken into consideration. Because, the power consumption of a platform not only depends on its microcontroller and/or radio but also on its auxiliary components and their quiescent current [21,22]. A comparison of the evaluated platforms equipped with Sensirion SHT15 humidity sensors in terms of power consumption is given in Table 1.

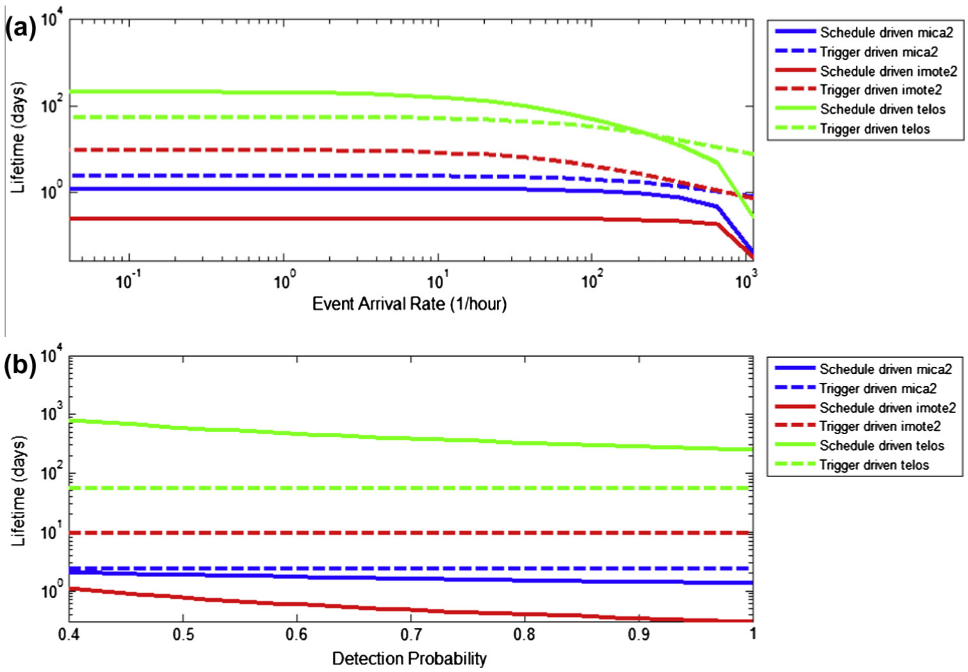

Fig. 2. (a) Duty cycle: 1, event inter-arrival rate: 1 s to 1 day. (b) Duty cycle: 0.2–0.8, event inter-arrival rate: 1 day.

Table 1
A comparison of the evaluated platforms in terms of power consumption [23].

	Mica2	Telos	Imote2
CPU on, sensors active	37.8 mW	7.2 mW	>100 mW
CPU on, radio off	36 mW	5.4 mW	>100 mW
CPU on, radio TX/RX	117 mW	58.5 mW	>150 mW
CPU in sleep mode, radio off	54 µW	15.3 µW	0.1 mW

In this figure, the relationship between sensor node lifetime and packet transmission strategies for different commercially available WSN node platforms is shown.

From Fig. 2, it is obvious that the frequent use of the radio transceiver, which is one of the characteristic features of the online SLAM applications, significantly shortens the battery lifetime of a sensor node. Hence, there is a trade-off here between the sensor lifetime and SLAM performance. From these simulation results, it can be seen that the Telos sensor nodes are more suitable, among the compared sensor nodes, for SLAM applications due to their longer battery lifetime and the low-power IEEE 802.15.4 protocol support. As a general result, for WSN-based CSLAM operations, low power and energy-efficient communication protocols need to be developed. Especially, since packet transmissions affect the battery lifetime significantly, packet losses leading to waste of battery lifetime should be prevented as much as possible.

To understand the relationship between battery lifetime and number of transmitted packets better, a field experiment using a pair of Telos sensor nodes was conducted. During this experiment, the transmitting node was sending packets to the receiving node with a constant rate of 2 packets per second. Until the battery voltage dropped down to 1.8 V, the cut-off voltage for the MSP430 MCU of the Telos node, the battery voltage decrease was continuously recorded.

As shown in Fig. 3, a nonlinear relationship between the battery voltage and the number of transmitted packets exists. As the number of transmitted packets increases, the battery voltage decreases with a different rate. Note that these observations are consistent with the results already reported in [24,25]. However, they are quite different from the results of simulation-based studies. This is due to the fact that real batteries' operation is more complex than the battery models assumed during simulations; the real ones typically exhibit non-ideal characteristics [24,25].

The results of the simulation studies and the experiment show that the realization of WSN-based CSLAM directly relies on the energy efficiency and the reliable communication capabilities of the system. In such a system, the number of transmitted packets directly affects the lifetime of a sensor node. However, communication requirements of Multi-Robot SLAM cannot be directly modeled based on sensor nodes' hardware since non-ideal characteristics of their batteries constitute a problem for battery-operated sensor nodes even if it is not a major problem for mobile robots. These observations confirm that, for the proposed approach, a reliable transport solution which includes efficient congestion detection and control mechanisms can alleviate the encountered problems.

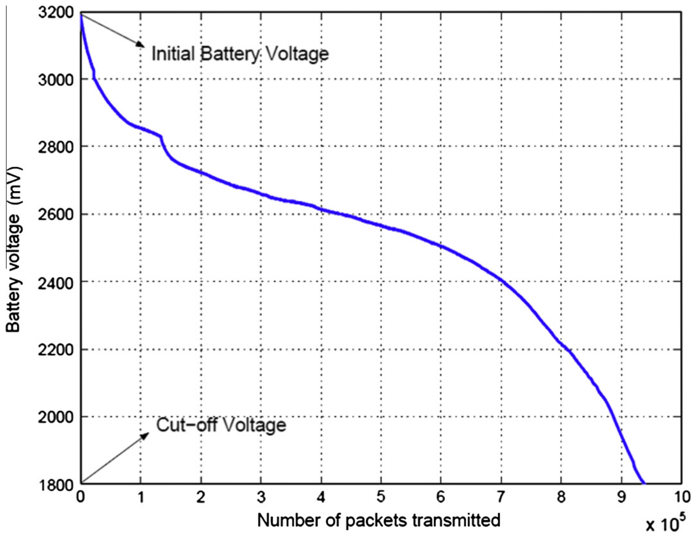

Fig. 3. Battery voltage vs. number of packets transmitted for a using a pair of communicating Telos sensor nodes.

4. Design considerations in WSN-based CSLAM applications

The SLAM-related performance evaluations conducted in this study were based on Extended Kalman Filtering (EKF). In principle, EKF uses a landmark-based map and operates recursively in two stages: Prediction and update [4,26]. During the prediction stage, the command $u(k)$ and the robot motion model are utilized to estimate the robot's location. Then, during the update stage, the new observation $z(k)$ from an exteroceptive sensor is used to update the landmark's position and to refine the estimation of the robot's location.

EKF consists of five steps as shown in Fig. 4. The five steps of each EKF iteration can be summarized as follows:

- *State prediction:* By using the motion model of the robot, an estimate of the robot position is generated.
- *Measurement prediction:* The fusion of the observation into the state estimate is accomplished by first calculating a predicted observation through the observation model.
- *Observation:* Observations are received from the robot's exteroceptive sensors which must be associated with particular features in the environment.
- *Matching:* Matched pairings are integrated in an iterative way as explained in [27,26].
- *Estimation:* After successful matching of observation and predictions, the innovations are calculated, and using these innovations, the posterior estimates of the robot position and associated covariance are computed [26].

The accuracy of SLAM depends on several factors such as the quality of sensors, maximum observation range, observation noises, time interval between observations, time interval between odometry readings, environmental factors, the SLAM

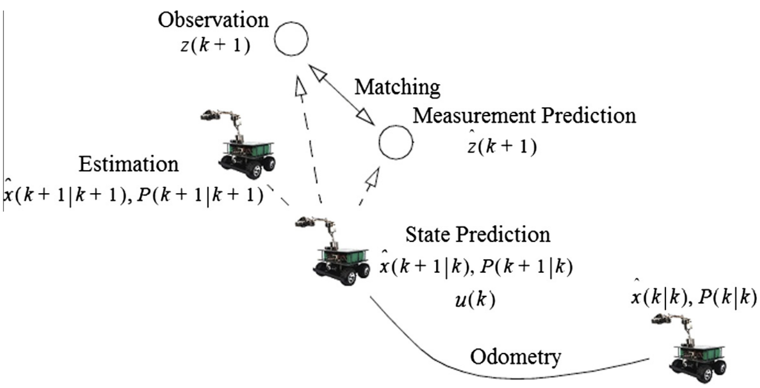

Fig. 4. The steps of EKF-based SLAM.

algorithm, and the data association algorithm. Though most SLAM benchmarking approaches use a metric relying on a global reference frame to compute the error, Kümmerle et al. [28] proposed a new framework for analyzing the results of a SLAM approach based on a metric which uses only relative relations between poses and does not rely on a global reference frame. Similar to the common strategy, the accuracy of SLAM can be evaluated based on the estimated trajectory of a robot. The difference between the absolute ground truth pose and the estimated pose of the robot is computed by using the mean squared error and this measurement over all ground truth points in time is referred to as the absolute trajectory error (ATE) [29]. ATE is given by:

$$error = \frac{1}{N_{GT}} \cdot \sum_{i=1}^{N_{GT}} \sqrt{(x_i - \tilde{x}_i)^2 + (y_i - \tilde{y}_i)^2} \quad (3)$$

where N_{GT} represents the number of ground truth measurements, x_i and y_i represent the absolute ground truth position at step i, and $\tilde{x}_i$ and $\tilde{y}_i$ represent the estimated pose of the robot at step i.

To show the effects of the observation range, speed, time interval between observations, and time interval between odometry readings on SLAM accuracy, a set of simulation studies were conducted. Fig. 5 shows the trajectory the robot followed in the simulation studies. Tables 2–6 show the different parameters related to SLAM and the obtained results in detail. These results can be summarized as follows:

- As listed in Tables 2 and 3, it is observed that increasing the observation range decreases ATE and increases SLAM accuracy.
- As listed in Table 4, it can be seen that reducing the number of observations increases ATE and reduces SLAM accuracy.
- As listed in Table 5, reducing the number of odometry readings increases ATE and reduces SLAM accuracy.
- As listed in Table 6, increasing robot's speed increases ATE and reduces SLAM accuracy.

Moreover, increasing the number of observations and odometry readings also increases the number of packets to be transmitted and reduces the battery-life of a sensor node in case of using a WSN as the communication platform in CSLAM. Hence, there is a trade-off between the frequency of data transmissions required to carry odometry readings and observations, and WSN lifetime.

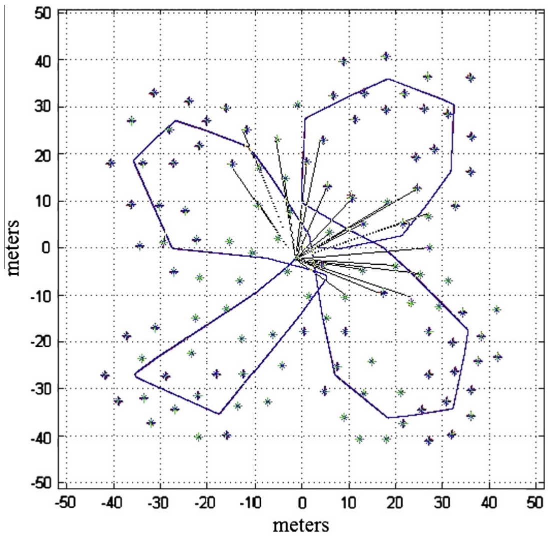

Fig. 5. Trajectory the robot followed. Blue crosses represent the landmarks in the environment. Green crosses represent the mapped landmarks. The black lines leaving a specific point, the robot, represent the process of obtaining sensory information. Blue lines represent the trajectory that the robot follows. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
ATE with different time intervals (odometry readings and observations) when observation range is 8 m.

Observation range (m)	Speed (m/s)	Time interval between odometry readings (s)	Time interval between observations (s)	ATE (m)
8	3	0.025	0.2	2.537002036
8	3	0.0375	0.3	6.126400652
8	3	0.05	0.4	10.62618905

Table 3
ATE with different time intervals (odometry readings and observations) when observation range is 30 m.

Observation range (m)	Speed (m/s)	Time interval between odometry readings (s)	Time interval between observations (s)	ATE (m)
30	3	0.025	0.2	2.30157509
30	3	0.0375	0.3	3.35098695
30	3	0.05	0.4	7.05382056

Table 4
ATE with different time intervals between observations.

Observation range (m)	Speed (m/s)	Time interval between odometry readings (s)	Time interval between observations (s)	ATE (m)
30	3	0.025	0.2	2.30157509
30	3	0.025	0.3	3.64752443
30	3	0.025	0.4	6.51582819

Table 5
ATE with different time intervals between odometry readings.

Observation range (m)	Speed (m/s)	Time interval between observations (s)	Time interval between odometry readings (s)	ATE (m)
30	3	0.2	0.025	2.30157509
30	3	0.2	0.04	2.80689160
30	3	0.2	0.05	3.37887262

Table 6
ATE with different speeds.

Observation range (m)	Time interval between odometry readings (s)	Time interval between observations (s)	Speed (m/s)	ATE (m)
30	0.025	0.2	1	0.28495157
30	0.025	0.2	3	2.30157509
30	0.025	0.2	4	10.8297489
30	0.025	0.2	7	13.3769785

5. Centralized and distributed merging methods in WSN-based CSLAM applications

In this section, the centralized and distributed map merging methods in WSN-based CSLAM are described and evaluated.

5.1. Centralized map merging

In this approach, each robot transmits its control input and observations to a central controller which runs a monolithic SLAM filter to estimate the positions of the robot and the landmarks. Therefore, the full covariance matrix must be updated with each prediction and observation of each robot [30]. Here, the central controller is responsible for building the global map. All sensor readings are sent to the central controller and the controller builds the global map. Global map consists of all robots and landmarks in the environment. Fig. 6 shows WSN-based CSLAM map merging. All sensor readings are carried through WSN channels to the central controller periodically. Wireless sensor nodes can communicate with each other and with the central controller within some distances. If a robot is far away from the central controller, then it cannot directly communicate with the controller. The arrows in Fig. 6 show the data flow of WSN-based CSLAM centralized map merging. Centralized map merging approach is feasible if the packet sizes of control inputs and observations are relatively small. Therefore, WSN-based CLAM can be successfully used if laser scanners, ultrasonic range finders or infrared range finders are used for exteroceptive sensing.

To show the effects of this approach, an evaluation environment using ns-2 [31] was developed. The parameters used in this study are listed in Table 7. Unless otherwise specified, these values were used in the simulations. In these simulations, 200 sensor nodes were randomly positioned in a $100 \times 100\ m^2$ deployment field. Parameters such as radio range and interface queue (IFQ) length were carefully chosen to mirror typical sensor mote values [32]. Event centers were randomly chosen and all nodes within the event radius behave as sources for that event. In the simulations, a Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA) based Medium Access Control (MAC) protocol was used. If the robots (shown as the green circles in Fig. 6) detect landmarks, they send the sensor measurements to the controller.

In both wired and wireless networks, CSMA is used to detect the absence of other traffic before transmitting on a shared transmission medium [33]. Different from Carrier Sense Multiple Access with Collision Detection (CSMA/CD) which uses the

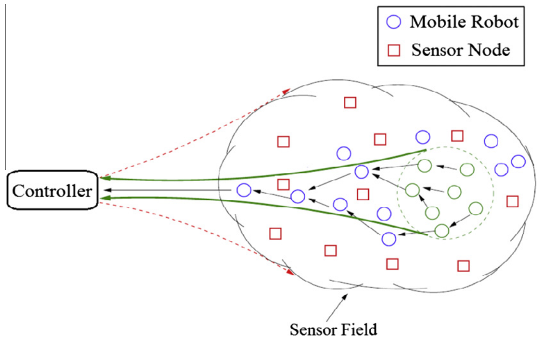

Fig. 6. WSN-based CSLAM centralized map merging. Blue and green circles represent the robots. However, the green circles represent the robots that transmit their sensory information to the controller. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 7
ns-2 simulation parameters.

Specification	Value
Area of sensor field	100×100 m^2
Number of sensor nodes	200
Radio range of a sensor node	40 m
Network density	100 neighbor/node
Packet length	30 bytes
IFQ length	65 packets
Transmit power	0.660 W
Receive power	0.395 W
Decision interval	1 s

terminating transmission to detect and deal with the collision and then prevents the collision happening again to improve the performance, CSMA/CA acts to reduce the probability of the first-time collision happen on the channel by checking if the channel is clear or not, once a node receives a packet. Regarding MAC, IEEE 802.15.4 standard allows two types of channel access mechanisms: beacon-enabled and non beacon-enabled. Non-beacon enabled networks use unslotted CSMA/CA. Each time a device needs to access the radio channel, it waits for a random backoff period and then senses the channel. If the channel is found to be idle, then the device transmits the data. Otherwise, if the channel is busy, it waits for another random period before trying to access the channel again. On the other hand, beacon-enabled networks use a slotted CSMA/CA channel-access mechanism, and a superframe structure managed by the personal area network (PAN) coordinator is implemented [33]. The superframe structure is used by IEEE 802.15.4 networks to provide guaranteed time slots for low-latency applications and applications requiring a specific data bandwidth [34].

CSMA/CA plays a critical role for event-driven WSNs. Event-driven WSNs are typically densely deployed for increased fault-tolerance, coverage and connectivity. Therefore several nodes try to access the channel at the same time during the occurrence of an event. This results in severe contention for channel access. To investigate this problem, in this subsection, two case studies are given. In these case studies, the relation between the application-specific requirements, map update frequency (i.e., channel access) and transmission delay, are shown. In Fig. 7, the sensor reading-to-controller delay distribution for different map update frequency, f, is shown. In this context, the time stamp field in the data packet header is used in order to compute the sensor reading-to-controller delay. As illustrated in Fig. 7, as the map update frequency increases, the sensor reading-to-controller delay distribution shifts to the right, i.e., the sensor reading-to-controller delay increases. This results from congestion caused by the increased network load. Since congestion and contention are closely related to each other, each one of them could potentially trigger the other. For example, sensor nodes not being able to access the channel due to contention can cause packets to rapidly queue up at each node and hence can lead to congestion. Similarly, increased congestion in the network can trigger channel access issues since each node now has a greater number of packets to transmit and hence attempts to access the channel more frequently.

In Fig. 8, the number of packets received in the controller with varying sensor reading-to-controller delay bounds is shown. As illustrated in Fig. 8, by increasing the map update frequency, the number of packets received in the controller meeting the corresponding sensor reading-to-controller delay bound also increases. However, after some map update frequency, i.e. around the frequency $f = 10$, the number of packets received in the controller meeting the corresponding sensor reading-to-controller delay bound decreases, e.g., the number of packets received in the controller for $f = 20$ is less than the number of packets received in the controller for $f = 10$. This is because at this point the network capacity is exceeded and hence congestion occurs, and the data packets at the forwarding sensor nodes begin to be dropped. Note that until the network capacity

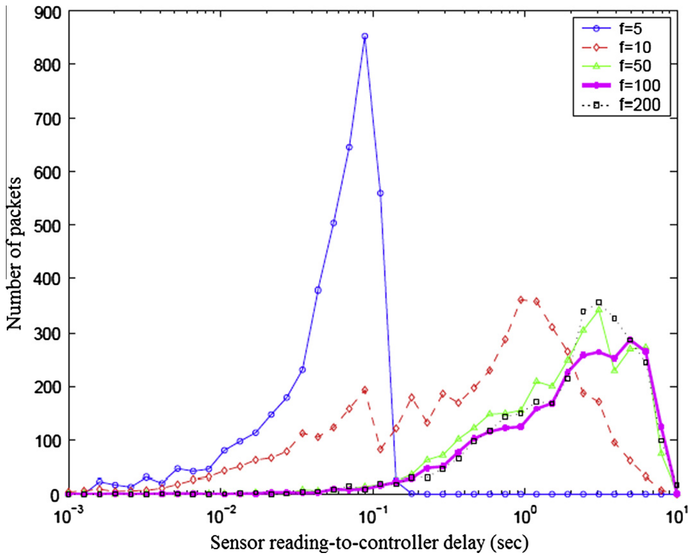

Fig. 7. Sensor reading to controller delay in a WSN-based CSLAM centralized map merging application for different map update frequencies, *f*.

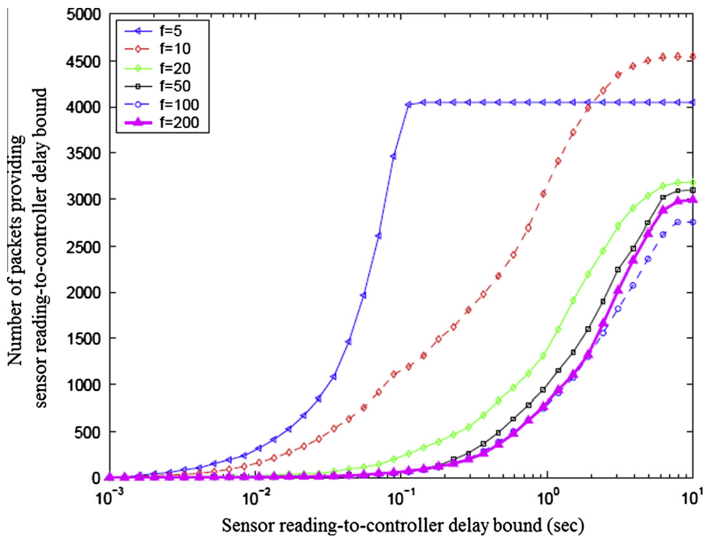

Fig. 8. The number of packets received in the controller in a WSN-based CSLAM centralized map merging application, for different sensor reading-to-controller delay bounds.

is reached, the number of packets received in the controller increases when the sensor reading-to-controller delay bound also increases. After the network capacity is reached, the number of packets received in the controller becomes constant.

In summary, when the application-specific delay bounds are considered, the number of received packets complying with delay bounds decreases significantly with the network congestion. These observations confirm the urgent need for efficient congestion detection and control mechanism in WSN-based CSLAM.

In addition to above-presented simulations, an extensive set of SLAM simulations was conducted in order to show the effects of packet drops on the performance of SLAM algorithms, since packet drops may increase uncertainty and may cause EKF to diverge. The data association algorithm used in these experiments was the Joint Compatibility Branch and Bound data association algorithm (JCBB) [13]. Fig. 9 shows the ground truth, i.e. the real positions of the landmarks and the real trajectory of the robot, used in the simulations. Fig. 10 presents the simulation results for the case that no packets are lost. Specifically, Fig. 10(a) shows the result of EKF-based SLAM, and Fig. 10(b) shows the vehicle errors (x, y, and theta). In a similar way, in Figs. 11–15 the corresponding simulation results are presented for the cases that different packets are lost. More precisely, Fig. 11 shows the simulation results when observation measurements are lost at step 21 of the EKF, Fig. 12 for the case that odometry readings are lost at step 21 of the EKF, Fig. 13 when observation measurements are lost at step 21 and step 22 of the EKF, Fig. 14 when odometry readings are lost at step 21 and step 22 of the EKF, and Fig. 15 shows the

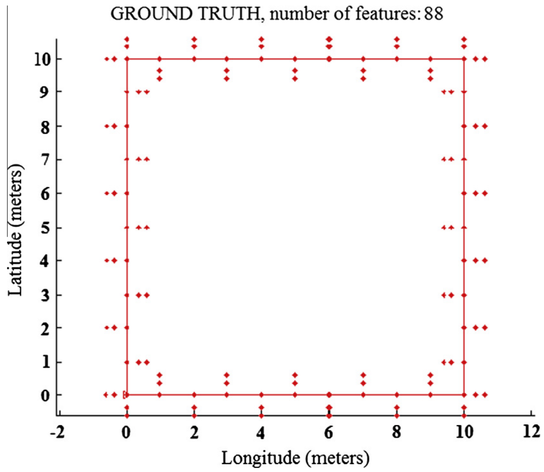

Fig. 9. The ground truth used for the WSN-based Multi-Robot SLAM centralized map merging evaluation. Red points represent the landmarks in the environment. The landmarks are specific important points in the map. The red line represents the trajectory that the robot follows. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

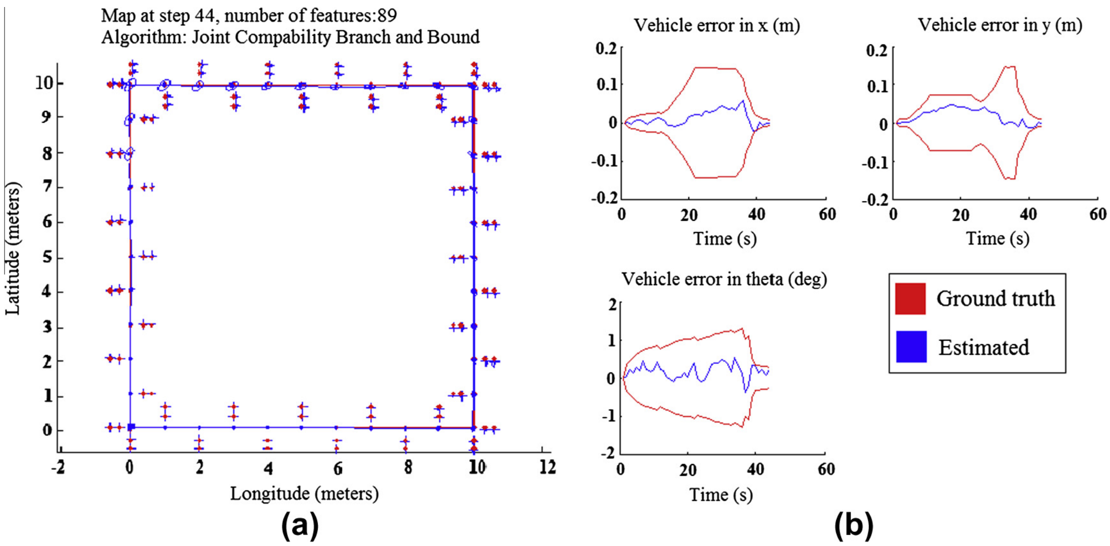

Fig. 10. (a) EKF-based SLAM Result, and (b) vehicle errors (x, y, theta) for the WSN-based Multi-Robot SLAM centralized map merging evaluation. Red points represent the landmarks in the environment. The landmarks are specific important points in the map. Blue crosses represent the locations of the landmarks that the robot thinks of. Blue circles represent uncertainty bounds. When the dimension of a blue circle is big, then the uncertainty at that point is high and the robot determines the location of the landmark with high error. The number of landmarks may change due to uncertainty. When there is high uncertainty and the deviation in the trajectory, the robot readdress a landmark which it has already seen. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

simulation results for the case that both observation measurements and odometry readings are lost at step 21 and step 22 of the EKF. Note that the selection of the steps where lost measurements exist was random. As it is evident by the simulation results presented in these figures, missing odometry readings and observation measurements increase vehicle errors, while reduce SLAM accuracy. Hence, the design of a lossless WSN is quite important for the success of CSLAM. Efficient solutions for missing data packets resulting from varying delays and packet losses should be developed as shown in the following section.

As a result, since some packet delays and drops may be experienced in this strategy, some solutions for this approach are as follows:

- *Subsampling of sensor measurements:* Evaluating the measurement model for a subset of all ranges, such as 12 of 360 laser range measurements, may give good results [35,3]. In this way, the data size and the frequency of observation data transmissions can be reduced. Since data payloads of sensor nodes are very small, data size to be transferred should be kept at minimum.

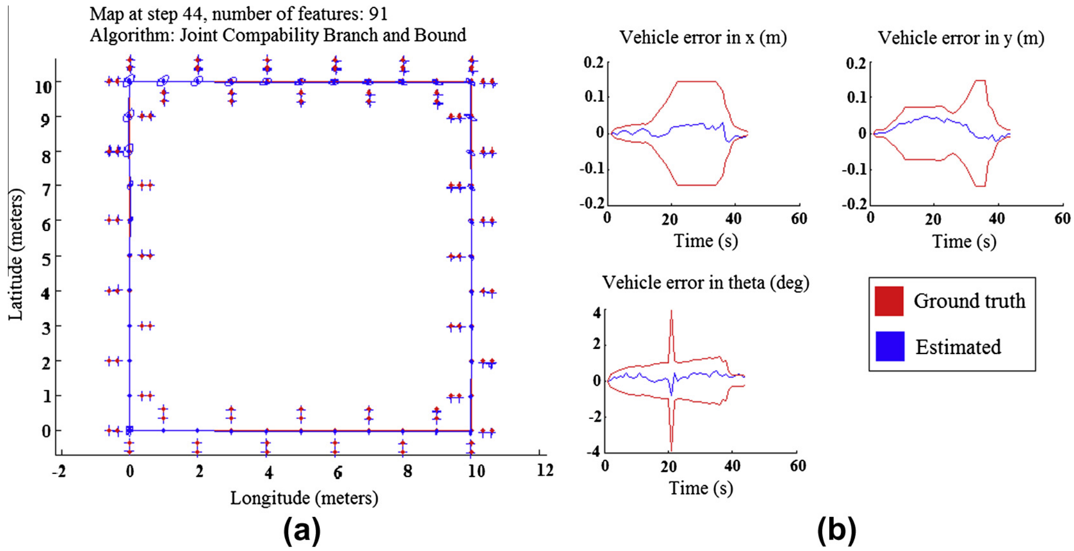

Fig. 11. (a) EKF-based SLAM Result and (b) vehicle errors (x, y, theta) with missing observation measurements at step 21 of the EKF. The interpretation of the employed symbols and colors can be found in Fig. 10 caption.

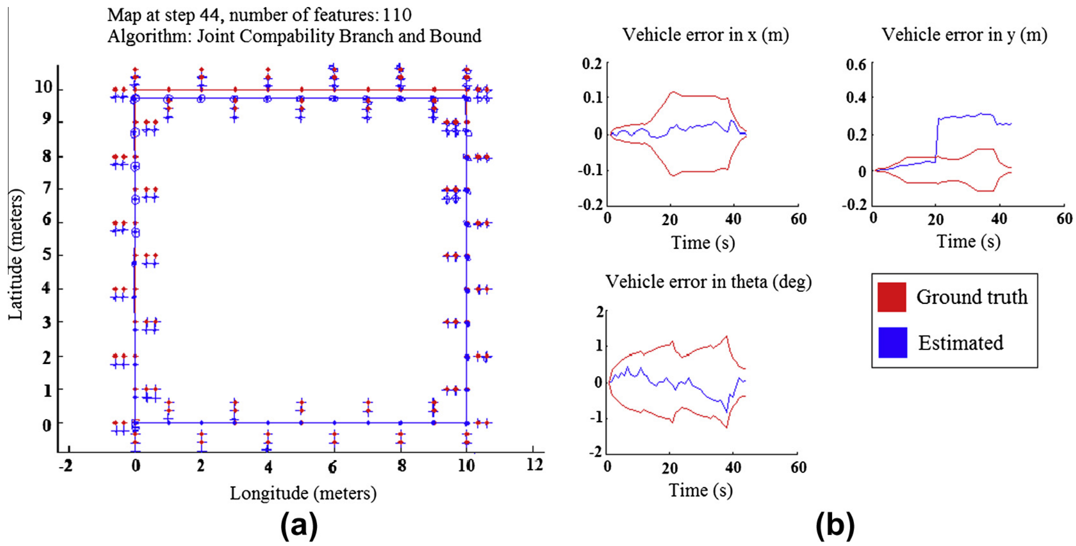

Fig. 12. (a) EKF-based SLAM Result and (b) vehicle errors (x, y, theta) with missing odometry readings at step 21 of the EKF. The interpretation of the employed symbols and colors can be found in Fig. 10 caption.

- *Delaying motion updates:* Instead of applying the motion update every time an odometry reading is obtained; Odometry readings over a short time period can be geometrically integrated and then applied [3]. This method can speed up the SLAM algorithm, while the central controller requires less frequent data transmissions.
- *Selective updating:* If occupancy-grid map representation is used, selective techniques may update a fraction of all grid cells. This method can speed up the SLAM algorithm.

5.2. Distributed map merging

In the distributed map merging approach, each robot builds an independent map of its local environment called sub-map. Periodically, these local sub-maps are matched, joined and fused together to form a global map [30]. Distributed map merging requires periodical map fusion. Therefore, each robot has to communicate with the other robots in the environment. In general, map merging is a difficult process, since there are statistically independent stochastic maps. It aims to convey the information of the two maps into a single fully consistent stochastic map [36]. A feature in one map is randomly selected and

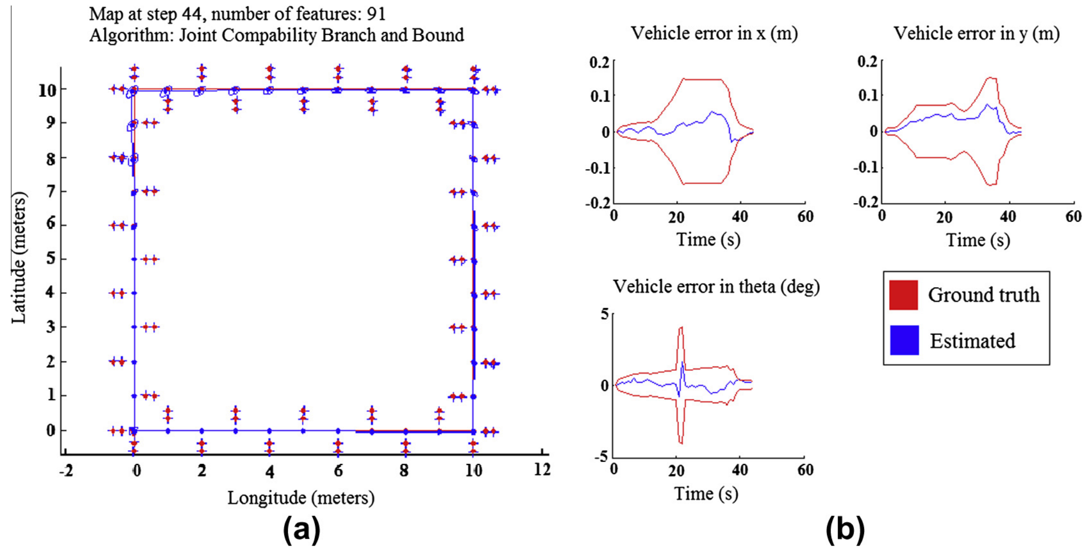

Fig. 13. (a) EKF-based SLAM Result and (b) vehicle errors (x, y, theta) with missing observation measurements at step 21, and 22 of the EKF. The interpretation of the employed symbols and colors can be found in Fig. 10 caption.

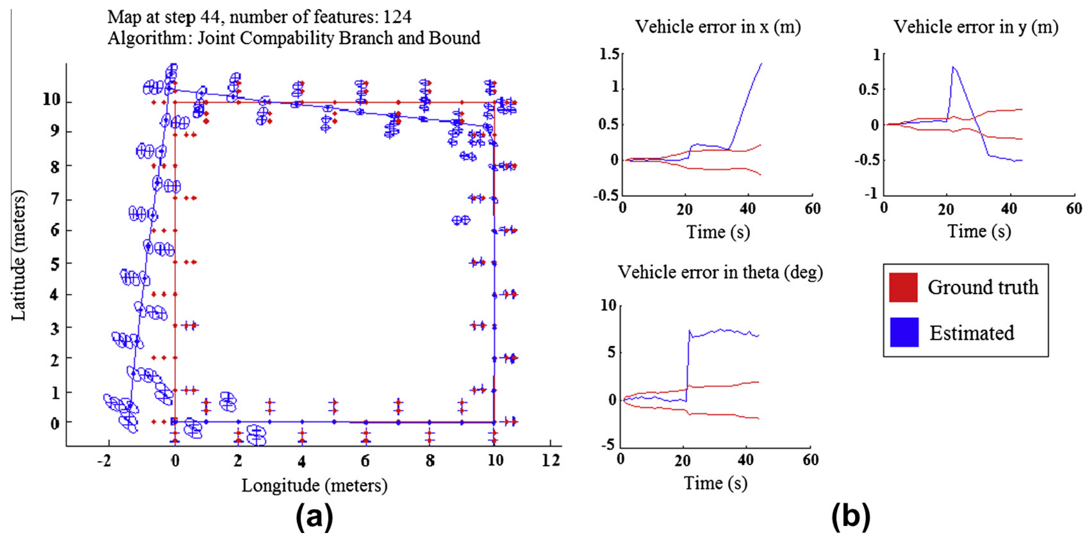

Fig. 14. (a) EKF-based SLAM Result and (b) vehicle errors (x, y, theta) with missing odometry readings at step 21, and 22 of the EKF. The interpretation of the employed symbols and colors can be found in Fig. 10 caption.

co-visible features of this feature are associated in the other map using methods such as Geometric Constraints Branch and Bound (GCBB) [36].

In WSNs, the data packets generated by the sensor nodes within the coverage of the sensed phenomenon are transported to the central controller in a multi-hop manner [37]. When two or more nodes attempt to transmit a frame at the same time, a collision occurs, and subsequently all frames get corrupted [38]. The standard mechanism to handle contention resolution problem is CSMA algorithms which basically attempt to break symmetries of failing transmissions being restarted at almost the same time by using randomized binary exponential backoff procedures. While wired nodes can listen during their own transmissions and employ CSMA/CD, nodes in wireless networks usually cannot listen to their own transmissions, and consequently colliding transmissions can only be detected after they have been completed. Thus wireless sensor nodes use CSMA/CA.

Because of the memory limitations of the sensor nodes and possible network load increase, congestion might be experienced in WSNs [38]. Congestion leads to both waste of communication and energy resources of the sensor nodes and also hampers the event detection reliability because of packet losses [39]. Hence, it is mandatory to address the congestion problem in the sensor field. In addition, in order to achieve an efficient congestion detection in WSN, the sensor nodes should be

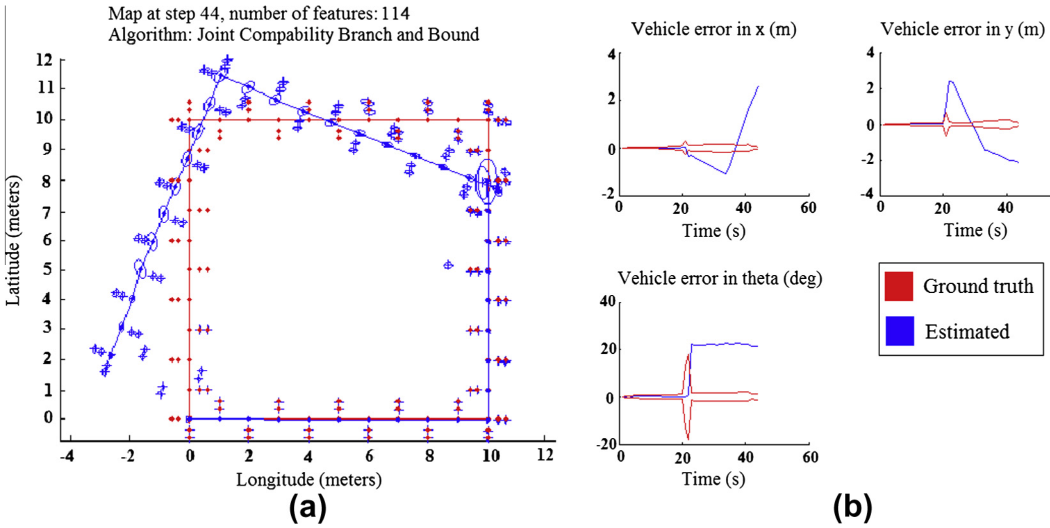

Fig. 15. (a) EKF-based SLAM Result and (b) vehicle errors (*x*, *y*, theta) with missing observation measurements and odometry readings at step 21, and 22 of the EKF. The interpretation of the employed symbols and colors can be found in Fig. 10 caption.

aware of the network channel condition around them, since the communication medium is shared and might be congested with the network traffic among other sensor nodes in the neighborhood [40,41]. Therefore, because of the shared communication medium nature of WSNs, the sensor nodes can experience congestion even when their buffer occupancy is small. On the other hand, contention increases collisions, which not only results in packet loss and decreased throughput, but also increases energy consumption and latency in the network. Collisions lead to back-off and piling up of packets in a node's transmissions queue, hampering its ability to transmit data. This results in network congestion and ultimately loss of event reports. Each collision also triggers a re-transmission, causing energy wastage and delayed event reporting. To investigate these interconnected problems, in this subsection, a case study is given.

In order to investigate the impact of the channel contention on the congestion level of the neighboring nodes, a simulation study using ns-2 [31] was performed. The network configuration is shown in Fig. 16, in which node 0 and 1 (sources) send data to node 4 and 5 (destinations), respectively.

During the time period between 4 and 6 s, the node 0 increases its transmission rate, which creates a hot spot around node 2. In Figs. 17 and 18, the resulting packet delay and buffer occupancy at the nodes 2 and 3 are shown, respectively. As it can be seen from Fig. 17, it is evident that both buffer occupancy ratio and average packet delay between 4 and 6 s were significantly increased at node 2; these metrics reflect the increased congestion level at node 2 during the specific time interval.

On the other hand, as shown in Fig. 18, even though the buffer occupancy at node 3 was small between 4 and 6 s (buffer occupancy ratio is almost 20%), the average packet delay was significantly increased during the specific time interval. This is because even when the incoming input traffic does not change during this time period, the increased channel contention around node 3 causes packet collisions and retransmissions resulting in increased packet delay. Note that, it is difficult to detect the level of congestion solely based on the buffer occupancy at node 3. Therefore, for the efficient congestion detection

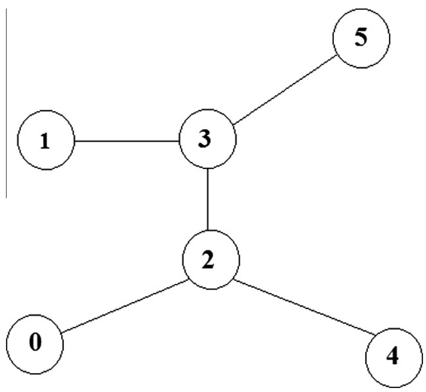

Fig. 16. A simple wireless ad hoc network of 6 nodes used to investigate the impact of the channel contention on the congestion level of the neighboring nodes. Only the nodes connected by a line are within each other's communication range.

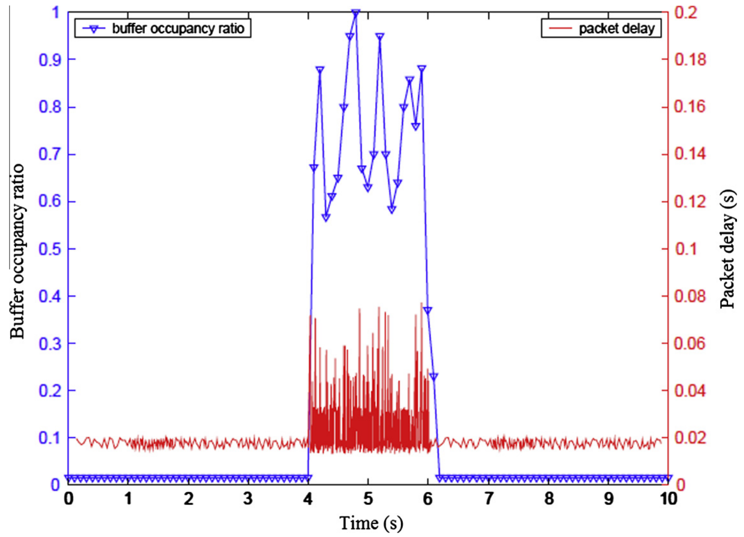

Fig. 17. Buffer occupancy ratio and packet delay observations at node 2 when node 0 increases its transmission rate (please also refer to Fig. 16).

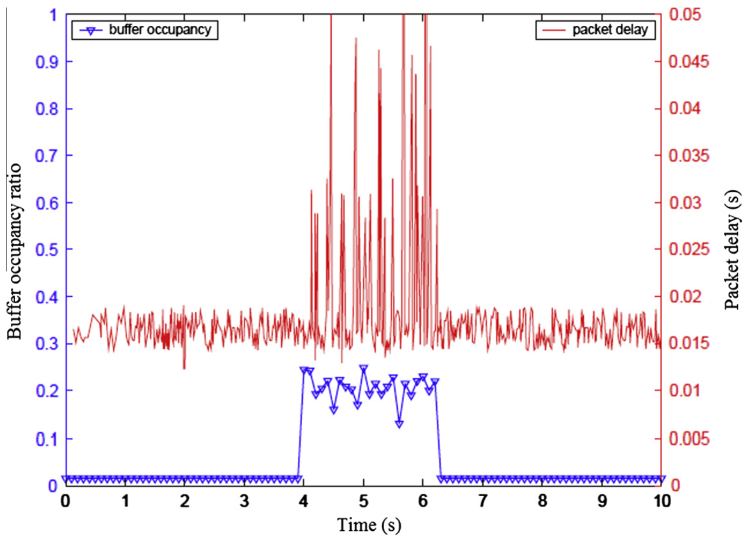

Fig. 18. Buffer occupancy ratio and packet delay observations at node 3 when node 0 increases its transmission rate (please also refer to Fig. 16).

in a WSN-based multi-robot system, a combined approach is required. In this regard, a combined congestion detection mechanism based on both average node delay calculation and local buffer level monitoring of the nodes is suggested here in order to detect congestion accurately in the network. Note that the average node delay at a node gives an idea for the contention around that node, i.e., how busy is the network vicinity of the node.

To show the relationship between the map size and processing time of each step, an extensive set of SLAM simulations was conducted for different multitude of merged maps. As already mentioned, the simulations were based on Joint Compatibility Branch and Bound data association algorithm (JCBB) [13]. Fig. 19(a) shows the ground truth, i.e. real positions of the landmarks and real trajectory of the robot, while Fig. 19(b) shows the result of EKF-based SLAM when a single map is used to store the positions of the landmarks and current robot position. Fig. 20(a) shows the processing time of each EKF step in the single map case, Fig. 20(b) shows the processing time of each EKF step when two maps are used to store the positions of the landmarks and current robot position. Fig. 21(a) shows the processing time of each EKF step when three maps are used to store the positions of the landmarks and current robot position, Fig. 21(b) shows the processing time of each EKF step when four maps are used to store the positions of the landmarks and current robot position.

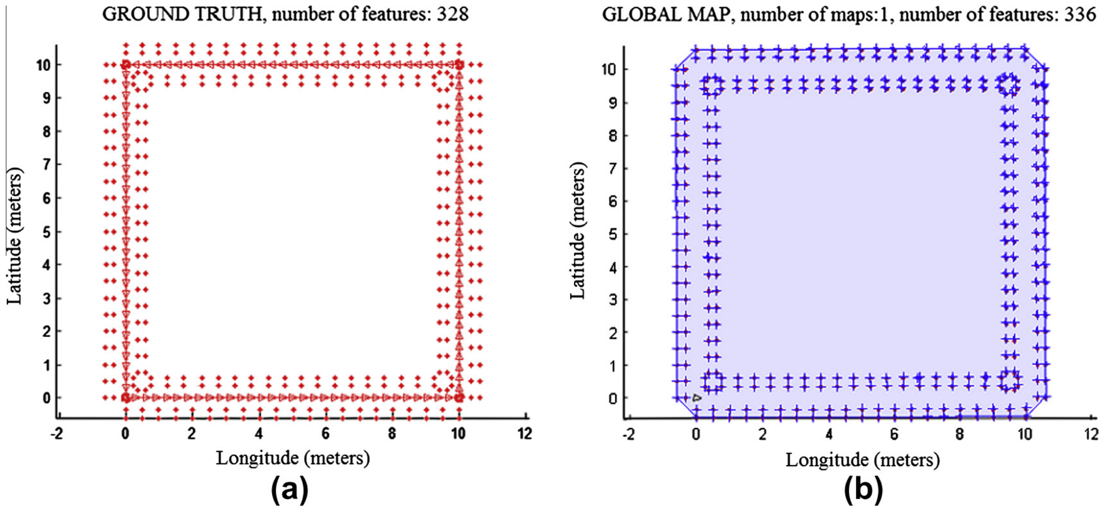

Fig. 19. (a) Ground truth. (b) Single map case when the number of maximum features: 400, total steps of EKF: 168. Red and blue points/crosses represent the landmarks in the environment. The landmarks are specific important points in the map. Red triangle (a) and the triangle (b) represent the robot. Solid blue line in (b) shows the borders of the region mapped. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

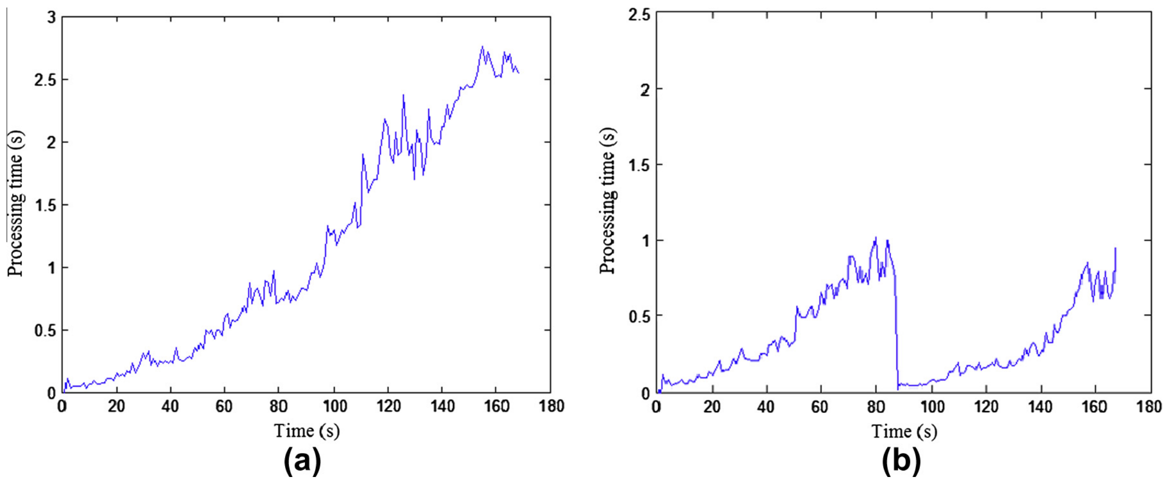

Fig. 20. (a) Processing time of each EKF step in single map case (we simulate the scenario that consists of a single robot). (b) Processing time of each EKF step in two map case (we simulate the scenario that consists of two robots).

These simulations show that limiting the number of features in a global map by dividing it into submaps, by using a team of mobile robots, decreases processing time required by each step. If there are a lot of natural and/or artificial features in an environment where CSLAM is performed, limiting the number of features in maps decreases processing time of each step. The trade-off between map accuracy and usability should be carefully considered. Also, instead of transmitting large sized map data at long intervals, small sized map data can be transmitted at short intervals. This method reduces the load on a WSN.

5.3. Discussion

Based on the preliminary results, in order to quantify the benefits of using WSNs for communication in CSLAM applications, a performance metric is needed. One of the metrics for such an evaluation is calculated by using (4) and is inversely proportional to the combined distance travelled by each robot, and directly proportional to the total area covered.

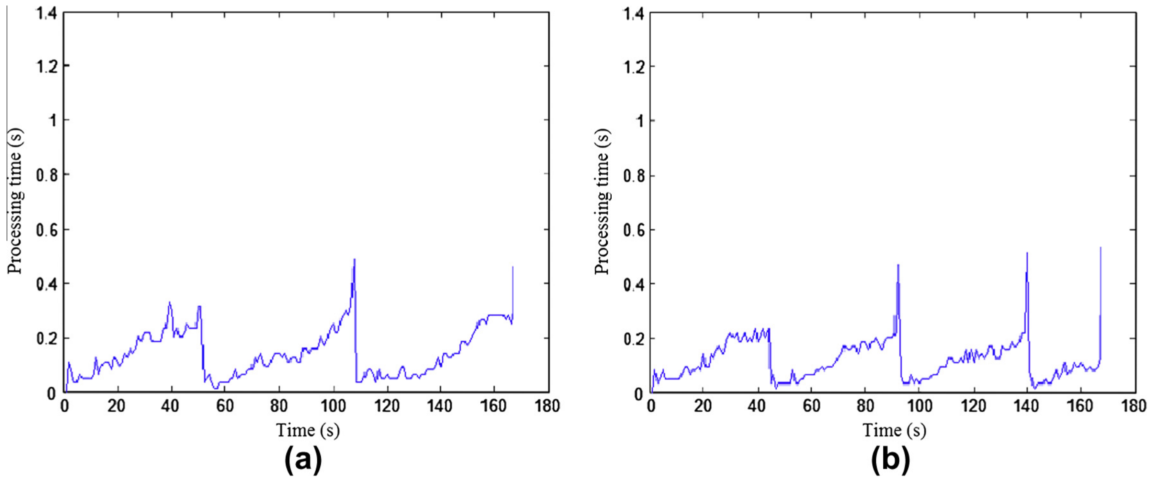

Fig. 21. (a) Processing time of each EKF step in three map case (we simulate the scenario that consists of three robots). (b) Processing time of each EKF step in four map case (we simulate the scenario that consists of four robots).

$$Q = \frac{A}{\sum_{i=1}^{n} d_i} \tag{4}$$

where d_i is the distance travelled by the robot i, A is the total area covered, and n is the total number of robots.

Though the use of WSNs for communication in CSLAM applications improves communication between robots and the control center for both the centralized map merging and the distributed map merging approaches, and thus address the common problem of outdoor SLAM applications and cooperative robotic exploration missions, WSNs are inherently unreliable and unpredictable, which has been the case for many of the sensor network deployments. However, recent approaches on radio diversity such as the one proposed by Kusy et al. [11] show that WSN links can be nearly reliable with multiple radio bands. This method can be utilized to deal with packet losses, and this makes the proposed strategy superior over other strategies.

In WSNs, one of the key factors is energy-efficiency. Though some techniques such as energy-harvesting alleviate the limited node/network lifetime problem, this must be handled properly. For this purpose, an energy-efficient framework to handle the trade-off between energy consumption and reliability is proposed in [42]. The framework prolongs the network lifetime and at the same time satisfies application-specific requirements. It basically involves an energy-aware adaptation module that captures the application's reliability requirements, and autonomously configures the MAC layer based on the network topology and the traffic conditions in order to minimize the power consumption. This framework will be integrated into our approach in the next steps of this ongoing study.

6. Conclusion

In this study, we have focused on the use of WSNs for communication in CSLAM applications and have explained its potential advantages and design challenges. In the proposed approach, WSNs provide a distributed communication topology to enable full communication between robots and improve the performance of CSLAM. The main benefits of the WSN-based communication architectures in CSLAM applications over the Wi-Fi based communication architecture is that greater coverage can be achieved at the same cost, since the cost of wireless sensor nodes is significantly less than the cost of mobile robots. Also, the WSN-based communication architecture can be used to monitor environmental conditions during CSLAM applications. This can offer many benefits especially in search and rescue operations.

We have conducted a comprehensive set of performance evaluations to prove the effectiveness of the proposed approach. The results of the performance evaluations show that WSNs can be effectively used to carry data packets of exteroceptive sensors with small-sized measurement data, such as laser range finders, ultrasonic range finders, and infrared range finders. Although our performance evaluations provide valuable insights into design issues for WSN-based CSLAM applications, they are only first steps. Future work consists of conducting field tests by using mobile robots and IEEE 802.15.4-based wireless sensor nodes and the use of sensor nodes as storage devices for local map fragments.

Acknowledgment

A preliminary shorter version of this paper was presented at the IEEE International Conference on Mechatronics, ICM 2011, April 13–15, 2011, Istanbul, Turkey [5].

References

[1] Dissanayake G, Newman P, Clark S, Durrant-Whyte HF, Csorba M. A solution to the simultaneous localization and map building (SLAM) problem. IEEE Trans Rob Autom 2001;17(3):229–41.
[2] Montemerlo M, Thrun S, Koller D, Wegbreit B. FastSLAM 2.0: an improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In: Proceedings of the 18th international joint conference on artificial intelligence (IJCAI'03); 2003. p. 1151–6.
[3] Thrun S, Burgard W, Fox D. Probabilistic robotics. Cambridge (USA): MIT Press; 2005.
[4] Williams SB. Efficient solutions to autonomous mapping and navigation problems. Ph.D. dissertation, University of Sydney; 2001.
[5] Tuna G, Gulez K, Gungor VC. Communication related design considerations of WSN-aided multi-robot SLAM. In: Proceedings of the 2011 IEEE international conference on mechatronics (ICM 2011); 2011. p. 493–8.
[6] Fenwick JW, Newman PM, Leonard JJ. Cooperative concurrent mapping and localization. In: Proceedings of the 2002 IEEE international conference on robotics and automation; 2002. p. 1810–7.
[7] Rodriguez-Losada D, Matia F, Jimenez A. Local maps fusion for real time multirobot indoor simultaneous localization and mapping. In: Proceedings of the 2004 IEEE international conference on robotics and automation, vol. 2; 2004. p. 1308–13.
[8] Thrun S, Liu Y, Koller D, Ng AY, Ghahramani Z, Durrant-Whyte H. Simultaneous localization and mapping with sparse extended information filters. Int J Rob Res 2004;23(7–8):693–716.
[9] Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E. Wireless sensor networks: a survey. Comput Netw 2002;38:393–422.
[10] Gungor VC, Hancke GP. Industrial wireless sensor networks: challenges, design principles, and technical approaches. IEEE Trans Ind Electron 2009;56(10):4258–65.
[11] Kusy B, Richter C, Hu W, Afanasyev M, Jurdak R, Brunig M, et al. Radio diversity for reliable communication in WSNs. In: Proceedings of the 2011 10th international conference on information processing in sensor networks (IPSN); 2011. p. 270–81.
[12] Aulinas J, Salvi J, Llado X, Petillot Y. Local map update for large scale SLAM. IET Electron Lett 2010;46(8):564–6.
[13] Paz LM, Tardos JD, Neira J. Divide and conquer: EKF SLAM in O(n). IEEE Trans Rob 2008;24(5):1107–20.
[14] Jung D, Teixeira T, Barton-Sweeney A, Savvides A. Model-based design exploration of wireless sensor node lifetimes. In: Proceedings of the fourth european conference on wireless sensor networks (EWSN 2007); 2011.
[15] Tuna G, Gungor VC, Gulez K. Wireless sensor networks for smart grid applications: a case study on link reliability and node lifetime evaluations in power distribution systems. Int J Distrib Sens Netw 2013;2013:1–11. http://dx.doi.org/10.1155/2013/796248 [Article ID 796248].
[16] Wang F, Liu J. Duty-cycle-aware broadcast in wireless sensor networks. In: Proceedings of the IEEE INFOCOM'09; 2009. p. 468–76. http://dx.doi.org/10.1109/INFCOM.2009.5061952.
[17] Jung D, Teixeira T, Savvides A. Sensor node lifetime analysis: models and tools. ACM Trans Sens Netw 2009;5(1). http://dx.doi.org/10.1145/1464420.146442 [Article No. 3].
[18] TinyOS Community Forum || Hardware Designs; 2011. <http://www.tinyos.net/scoop/special/hardware>.
[19] Crossbow Technology; 2011. <http://www.xbow.com/>.
[20] Imote2 – TinyOS Documentation Wiki; 2011. <http://docs.tinyos.net/index.php/Imote2>.
[21] Polastre J, Szewczyk R, Culler D. Telos: enabling low power wireless sensor network research. In: Proc of IPSN/SPOTS; 2005. p. 364–9.
[22] Nachman L, Huang J, Shahabdeen J, Adler R, Kling R. IMOTE2: serious computation at the edge. In: Proc of international wireless communications and mobile computing conference; 2008. p. 1118–23.
[23] Hülsmann Michael, Windt Katja, editors. Understanding autonomous cooperation and control in logistics. Springer-Verlag; 2007.
[24] Gungor VC, Sastry C, Song Z, Integlia R. Resource-aware and link-quality-based routing metric for wireless sensor and actor networks. In: Proceedings of IEEE ICC 2007; 2007. p. 3364–9.
[25] Rao R, Vrudhula S, Rakhmatov DN. Battery modeling for energy-aware system design. IEEE Comput 2003;36(12):77–87.
[26] Tomatis N. Hybrid, metric-topological, mobile robot navigation. Ph.D. dissertation, Département de Microtechnique, École Polytechnique Fédérale de Lausanne; 2001.
[27] Perez JA, Castellanos JA, Montiel JMM, Neira J, Tardos JD. Continuous mobile robot localization: vision vs. laser. In: Proceedings of the IEEE international conference on robotics and automation, vol. 4; 1999. p. 2917–23.
[28] Kümmerle R, Steder B, Dornhege C, Ruhnke M, Grisetti G, Stachniss C, et al. On measuring the accuracy of SLAM algorithms. Auton Rob 2009;27(4):387–407.
[29] Ruckert EA. Simultaneous localisation and mapping for mobile robots with recent sensor technologies. M.S. thesis, Graz University of Technology; 2009.
[30] Williams SB, Dissanayake G, Durrant-Whyte H. Towards multi-vehicle simultaneous localisation and mapping. In: Proceedings of the IEEE int conf robotics automation; 2002. p. 2743–8.
[31] The Network Simulator – ns-2; 2010. <http://www.isi.edu/nsnam/ns/index.html>.
[32] MICA Motes and Sensors; 2010. <http://www.xbow.com/Products/WirelessSensorNetworks.htm>.
[33] Buratti C, Verdone R. Performance analysis of IEEE 802.15.4 non beacon-enabled mode. IEEE Trans Veh Technol 2009;58(7):3480–93.
[34] IEEE std 802.15.4-2996, Part 15.4. Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (WPANs); 2006.
[35] Stachniss C. Robotic mapping and exploration. Berlin (Germany): Springer-Verlag; 2009.
[36] Neira J, Tardos JD, Castellanos JA. Linear time vehicle relocation in SLAM. In: Proceedings of the IEEE international conference on robotics and automation; 2003. p. 427–33.
[37] Zhu C, Zheng C, Han G. A survey on coverage and connectivity issues in wireless sensor networks. J Netw Comput Appl 2012;35:619–32.
[38] Gungor VC, Vuran MC, Akan OB. On the cross-layer interactions between congestion and contention in wireless sensor and actor networks. Ad Hoc Netw 2007;5:897–909.
[39] Akan OB, Akyildiz IF. ESRT: event-to-sink reliable transport for wireless sensor networks. IEEE/ACM Trans Netw 2005;13(5):1003–16.
[40] Hull B, Jamieson K, Balakrishnan H. Techniques for mitigating congestion in sensor networks. In: Proc ACM SENSYS 2004; 2004. p. 134–47.
[41] Ramaboli AL, Falowo OE, Chan H. Bandwidth aggregation in heterogeneous wireless networks: a survey of current approaches and issues. J Netw Comput Appl 2012;35:1674–90.
[42] Francesco MD, Anastasi G, Conti M, Das SK, Neri V. Reliability and energy-efficiency in IEEE 802.15.4/ZigBee sensor networks: an adaptive and cross-layer approach. IEEE J Sel Areas Commun 2011;29(8):1508–24.

Gurkan Tuna is an Assistant Professor at the Department of Computer Programming of Trakya University (2006–....). He has authored several papers in international conference proceedings and refereed journals, and has been actively serving as a reviewer for international journals and conferences. His current research interests include smart grid, ad hoc and sensor networks, robotic sensor networks, multirobot simultaneous localization and mapping, and cooperative exploration strategies, and multisensor fusion.

Vehbi Çağrı Güngör received the Ph.D. degree in electrical and computer engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2007 after working at the Broadband and Wireless Networking Laboratory. Currently, he is an Associate Professor and Department Chair of Computer Engineering at Abdullah Gül University, Kayseri, Turkey. His current research interests are in smart grid communications, cloud computing, next-generation wireless networks, wireless ad hoc and sensor networks, cognitive radio networks, and IP networks.

Stelios M. Potirakis is an Assistant Professor at the Electronics Engineering Department of TEI of Piraeus (2009–. . ..). Experience: ''Audio'' magazine (1997–1999), ''Analog Electronics & Electroacoustics'' team of Intracom Defense Electronics (2000–2009). Research interests: analog electronics, analog circuit analysis, systems' modeling, complex systems' signal analysis, electroacoustics, applied acoustics, acoustic ecology, distributed sensor networks, characterization of smart/multi-function materials, and educational technologies.