Available online at www.sciencedirect.com
SCIENCE DIRECT®

ELSEVIER

Ad Hoc Networks 2 (2004) 351–367

Ad Hoc Networks

www.elsevier.com/locate/adhoc

Wireless sensor and actor☆ networks: research challenges☆☆

Ian F. Akyildiz, Ismail H. Kasimoglu *

Broadband and Wireless Networking Laboratory, School of Electrical and Computer Engineering, Georgia Institute of Technology, 250 14th Street, NW Room 556, Atlanta, GA 30332, USA

Received 15 January 2004; accepted 14 April 2004
Available online 19 May 2004

Abstract

Wireless sensor and actor networks (WSANs) refer to a group of sensors and actors linked by wireless medium to perform distributed sensing and acting tasks. The realization of wireless sensor and actor networks (WSANs) needs to satisfy the requirements introduced by the coexistence of sensors and actors. In WSANs, sensors gather information about the physical world, while actors take decisions and then perform appropriate actions upon the environment, which allows a user to effectively sense and act from a distance. In order to provide effective sensing and acting, coordination mechanisms are required among sensors and actors. Moreover, to perform right and timely actions, sensor data must be valid at the time of acting. This paper explores sensor-actor and actor-actor coordination and describes research challenges for coordination and communication problems.

Keywords: Wireless sensor and actor networks; Wireless sensor networks; Ad-hoc networks; Coordination; Real-time communication; Transport; Routing; MAC; Cross-layering

1. Introduction

Recent technological advances have lead to the emergence of distributed wireless sensor and actor networks (WSANs) which are capable of observing the physical world, processing the data, making decisions based on the observations and performing appropriate actions. These networks can be an integral part of systems such as battlefield surveillance and microclimate control in buildings, nuclear, biological and chemical attack detection [2], home automation [19] and environmental monitoring.

☆ We refer to entities that can act on the network as *actors* They are sometimes referred to as *actuators* in related literature.

☆☆ This work is supported by the National Science Foundation under contract ECS-0428329.

* Corresponding author. Tel.: +1 404 894 5141; fax: +1 404 894 7883.
E-mail addresses: ian@ece.gatech.edu (I.F. Akyildiz), ismailhk@ece.gatech.edu (I.H. Kasimoglu).

1570-8705/$ - see front matter
doi:10.1016/j.adhoc.2004.04.003

For example, in the case of a fire, sensors relay the exact origin and intensity of the fire to water sprinkler actors so that the fire can easily be extinguished before it becomes uncontrollable. Similarly, motion and light sensors in a room can detect the presence of people and then command the appropriate actors to execute actions based on the pre-specified user preferences.

In WSANs, the phenomena of sensing and acting are performed by sensor and actor nodes, respectively. Sensors are low-cost, low power devices with limited sensing, computation, and wireless communication capabilities. Actors are resource rich nodes equipped with better processing capabilities, higher transmission powers and longer battery life. Moreover, the number of sensor nodes deployed in a target area may be in the order of hundreds or thousands where such a dense deployment is usually not necessary for actor nodes, since actors have higher capabilities and can act on large areas.

WSANs have the following unique characteristics:

- *Real-time requirement:* In WSANs, depending on the application there may be a need to rapidly respond to sensor input. For instance, in a fire application, actions should be initiated on the event area as soon as possible. Moreover, the collected and delivered sensor data must still be valid at the time of acting. For example, if sensors detect a malicious person in an area and transmit this information to the disposer of a tranquilizing gas actors that person must then still be in the same area when actors carry out the task. Therefore, the issue of real-time communication is very important in WSANs.
- *Coordination:* Unlike WSNs where the central entity (i.e., sink) performs the functions of data collection and coordination, in WSANs, new networking phenomena called sensor-actor and actor–actor coordination may occur (see Section III). In particular, sensor-actor coordination provides the transmission of event features from sensors to actors. After receiving event information, actors need to coordinate with each other in order to make decisions on the most appropriate way to perform the action.

Many protocols and algorithms have been proposed for WSNs in recent years [2]. However, since the above listed requirements impose stricter constraints, they may not be well-suited for the unique features and application requirements of WSANs. Moreover, although there has been some research effort related to WSANs, to the best of our knowledge, none of the existing studies to date investigate research challenges occurring due to the coexistence of sensors and actors.

For example, both in [6] and [13] control engineering problems and existing technologies about sensor and actor networks are presented, respectively. However, neither of these studies investigates the interaction among sensors and actors. In [10], only actor–actor coordination is handled without any insight into the sensor-actor coordination problem. A TDMA MAC protocol is introduced in [4] where it is assumed that sensor and actor nodes are of same type which obviously does not reflect the actual WSANs. In [15], the routing problems are investigated between sensor and actor nodes. However, no coordination problems in sensor-actor or in actor–actor communications are considered in the study.

As a result, despite some existing research in WSAN, coordination and communication problems that arise in WSANs due to the coexistence of sensors and actors are yet to be investigated.

The remainder of the paper is organized as follows: In Section 2, we present the physical architecture of WSANs. We explain the requirements of sensor-actor and actor–actor coordinations in Sections 3 and 4, respectively. In Section 5, we investigate the protocol stack of nodes and corresponding challenges both for sensor-actor and actor–actor communications. Finally, the paper is concluded in Section 6.

2. Physical characteristics of WSANs

In WSANs, the roles of sensor and actor nodes are to collect data from the environment and perform appropriate actions based on this collected data, respectively. Thus, as shown in Fig. 1 these nodes are scattered in the *sensor/actor field* while the sink monitors the overall network and commu-

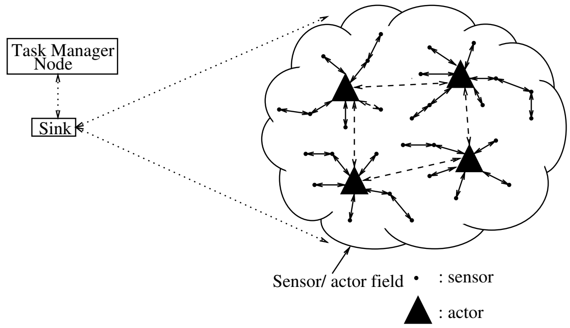

Fig. 1. The physical architecture of WSANs.

nicates with the task manager node and sensor/actor nodes.

Sensors detecting a phenomenon either transmit their readings to the actor nodes which process all incoming data and initiate appropriate actions, or route data back to the sink which may issue action commands to actors. We call the former case as *Automated Architecture* due to the non-existence of central controller, e.g., human interaction, while we call the latter case as *Semi-Automated Architecture* since the sink (central controller) collects data and coordinates the acting process. These two architectures are given in Fig. 2.

Depending on the types of applications, one of these architectures may be used. The advantage of the *Semi-Automated Architecture* is that it is similar to the architecture already used in wireless sensor network applications [2]. Thus, there is no need to develop new algorithms and protocols to perform communication and coordination.

In the remainder of the paper, we focus on the *Automated Architecture* because of:

1. *Low latency*: The sensed information is conveyed from sensors to actors, since they may be close to each other as shown in Fig. 1. As a result, the latency is minimized in the Automated Architecture.
2. *Long network lifetime*: In the *Semi-Automated Architecture*, as seen in Fig. 2(b), wherever the event occurs, event information always passes through the sensor nodes which are within one hop from the sink. Thus, those sensor nodes have excessive burden of relaying. When these nodes fail, the connectivity can be lost and the network can become useless. Although data aggregation techniques decrease the probability of these occurrences, sensor nodes around the sink are still more likely to fail than the other nodes in the network.
Similarly, in the *Automated Architecture*, as seen in Fig. 2(a), the nodes within one hop from the actors may have a higher load of relaying packets. However, here it is much more likely that for each event different actors may be triggered. This implies that relaying sensor nodes will also be different for each event. In other words, the relay load gets (more or less) evenly distributed between all nodes. As a result, the *Automated Architecture* will have longer lifetime than the *Semi-Automated Architecture*. Moreover, in the Automated Architecture since event information is transmitted locally through sensor nodes around the event area, sensors that are far from the event area do not function as relaying nodes, which results in network resource (i.e., energy and bandwidth, etc.) savings in WSANs.

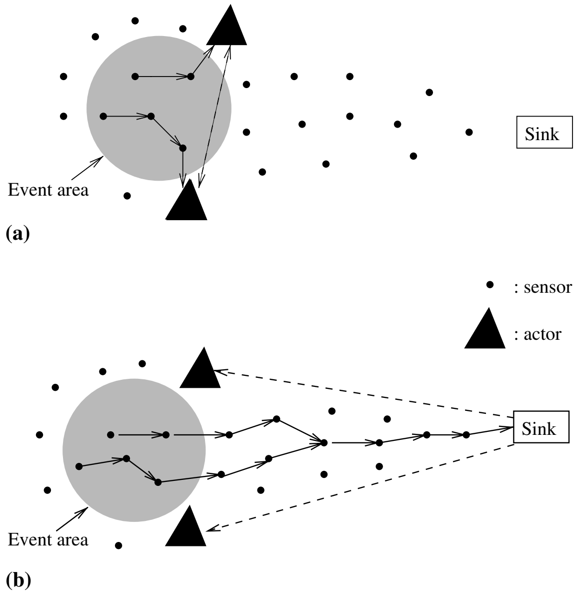

Fig. 2. (a) Automated vs. (b) Semi-Automated Architecture.

The components of sensor and actor nodes used in the WSAN applications can be seen in Fig. 3(a)

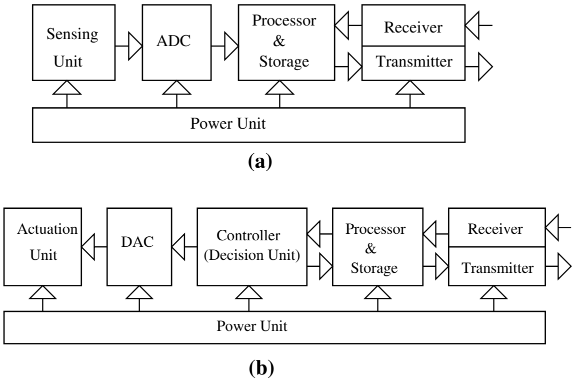

Fig. 3. The components of (a) sensors and (b) actors.

and (b), respectively. Sensor nodes are equipped with power unit, communication subsystems (receiver and transmitter), storage and processing resources, Analog to Digital Converter (ADC) and sensing unit, as shown in Fig. 3(a). The sensing unit observes phenomena such as thermal, optic or acoustic event. The collected analog data are converted to digital data by ADC and then are analyzed by a processor and then transmitted to nearby actors.

The decision unit (controller) functions as an entity that takes sensor readings as input and generates action commands as output. These action commands are then converted to analog signals by the Digital to Analog Converter (DAC) and are transformed into actions via the actuation unit as shown in Fig. 3(b).

In some applications, integrated sensor/actor nodes may replace actor nodes. Since an integrated sensor/actor node is capable of both sensing and acting, it has sensing unit and ADC in addition to all components of an actor node shown in Fig. 3(b).

One of the examples for an integrated sensor/actor node is a robot. However, a single robot may not have a sufficient sensing capability to sense the entire event area. Hence, in order to initiate more reliable actions, robots (integrated sensor/actor nodes) should act based on its own sensor readings as well as on the other nearby sensor nodes' data in the network. In other words, sensors transmit their readings to the nearby robots which process all sensor readings including their own sensor data. This way robots can collaborate with sensor nodes which provide them to have a reliable knowledge about the overall event. Then, the decision unit takes appropriate decisions and the actuation unit performs actions as in an actor node.

The use of integrated sensor/actor or actor node does not influence the overall architecture of WSANs. However, in most of the real applications, integrated sensor/actor nodes, especially robots, are used instead of actor nodes.

The robots designed by several *Robotics Research Laboratories* are shown in Fig. 4(a)–(d). Low-flying helicopter platform shown in Fig. 4(a) provides ground mapping, and air-to-ground cooperation of autonomous robotic vehicles [24]. However, it is likely that in the near future more several actuation functionalities such as water sprinkling or disposing of a gas can be supported by this helicopter platform, which will make WSANs much more efficient than today. An example of *Robotic Mule* which is called autonomous battlefield robot designed for the Army is given in Fig. 4(b). There are several autonomous battlefield robot projects sponsored by Space and Naval Warfare Systems Command [9] and Defense Advanced Research Projects Agency (DARPA) [8]. These developed battlefield robots can detect and mark mines, carry weapons, function as tanks or maybe in the future totally replace soldiers in the battlefield. Moreover, SKITs shown in Fig. 4(c) are networked tele-robots having a radio turret which enables communication over UHF frequencies at 4800 kbits/sec [22]. These robots can coordinate with each other by exploiting their wireless communication capabilities and perform the tasks determined by the application. Finally, possibly the world's smallest autonomous untethered robot (1/4 cubic inch and weighing less than an ounce) being developed in *Sandia National Laboratories* [20] is given in Fig. 4(d). Although it is not capable of performing difficult tasks that are done with much larger robots yet, it is very likely that it will be the robot of the future. A sensor node and a sink are given in Fig. 5. MICA is an open-source hardware and software platform that combines sensing, communications, and comput-

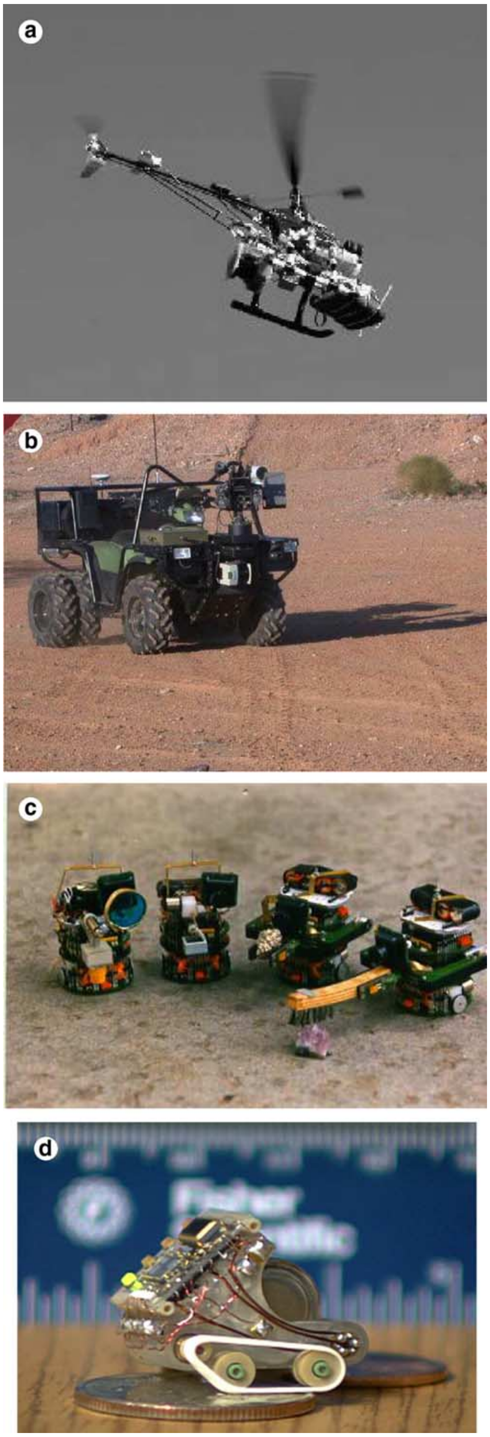

Fig. 4. Examples of robots: (a) Aerial mapping helicopter, (b) Robotic Mule, (c) Sub-kilogram intelligent tele-robots (SKITs) and (d) Mini-robot.

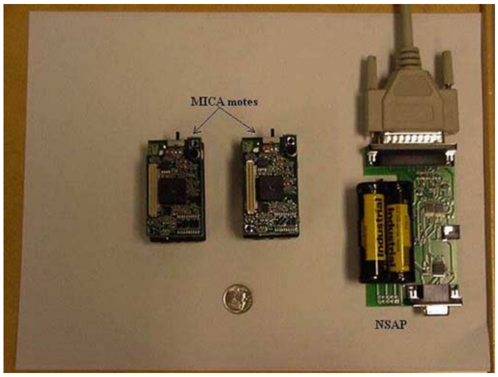

Fig. 5. Sensor nodes.

ing into a complete architecture to form an integrated wireless smart sensor.

In addition to sensor-actor communication, in most situations actor–actor communication is also required to achieve the overall application objective in WSANs. Since actors are resource-rich nodes with high transmission power, actor–actor communication can be long-range unlike sensor-actor communication. Furthermore, actor–actor communication is similar to the communication paradigm of ad-hoc networks due to the small number of (mobile) resource-rich actor nodes being loosely deployed. Therefore, WSAN can be considered as the union of wireless sensor and ad-hoc networks. In addition to both sensor and ad-hoc network challenges, there exist challenges due to the real-time properties and nature of "acting" phenomenon. In Section 3, we describe the characteristics and challenges of sensor-actor coordination which only deals with the transmission of event features to actors. In Section 4, we investigate the characteristics and challenges of actor–actor coordination which deals with the actions performed by actors after receiving event information.

3. Effective sensor-actor coordination

The most important characteristic of sensor-actor communication is to provide low communication delay due to the proximity of sensors and actors. The main problems which should be investigated for the sensor-actor coordination are:

1. *What are the requirements of this communication*?
2. *Which sensors communicate with which actors*?
3. *How is this communication realized*?

In addressing the first problem, one of the main requirements of sensor-actor communication is to consume low energy as in WSNs. Moreover, in some applications such as in fire, the communication traffic is typically delay sensitive. Therefore, another main requirement of sensor-actor communication is to support real-time traffic. To the best of our knowledge, to date no efficient solutions exist for real-time communications in WSANs.

An additional requirement for communication in WSANs is the need to ensure ordering of event data reported to the actors. For example, if there are two sensors reporting two different events to an actor or some actors in overlapping regions, then the reporting of those events must be done in the sequence in which the events were detected so that the correctness of the actions on the environment is guaranteed. We refer to this as the *ordered delivery* of information collected by the sensors.

Another important consideration is that if there are multiple sensors reporting an event, then the information from different sensors may arrive at the concerned actors approximately at the same time. This may be necessary to ensure that the action is performed once and in the entire event region. For example, if we consider a military application where sensors are deployed to detect enemy troops and actors to destroy them and the event being the presence of troops in a large area, then if the action is not performed in the entire region at once, it is conceivable that the troops would get alerted and hence disperse in different directions making it impossible to destroy the entire enemy troop.

While it may be desirable that this synchronization of event execution is done by actor–actor coordination as we describe in Section 4, it is also conceivable that the sensors can enable this synchronization. In some applications, where the event takes place in different locations, such as in the example given above, it might also be necessary that the events are passed on to the set of actors not necessarily close to or within the event area when the event was detected but to closest set of actors to the event when it is reported to them. In such cases, the sensors must be able to *track* the event and use this information to determine the set of actor(s) to send the information.

Therefore, new protocols must be developed for WSANs with the following objectives:

- provide real-time services with given delay bounds, according to application constraints,
- ensure an energy efficient communication among sensors and actors,
- ensure ordering between the different events when they are reported to the actors,
- provide synchronization among different sensors reporting the same event to multiple or same actor in order to facilitate a one-time response in the entire region,
- track and report the sensed phenomena to a different set of actors not necessarily based on proximity or energy limitations for the case when the events take place in different locations.

The developed protocols satisfying the above requirements of sensor-actor communication should also deal with the second and third problems stated above, that is, the sources/destinations involving in the transmission of sensor data as well as the type of this transmission.

For the sources/destinations involved in transmission reception of sensor data, there are four alternatives, namely:

- minimal set of actors to cover the event region or
- the minimum number of sensors to report the sensed event or
- both cases above or
- the entire set of actors and sensors in the vicinity of the region.

We refer to the first three classifications as the *redundancy elimination* problem in WSANs. This can be done to minimize the average power consumption of all the sensors and actors that are pre-

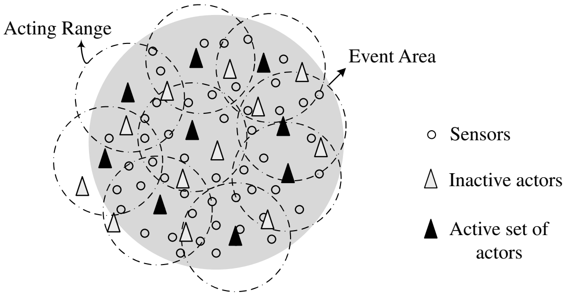

Fig. 6. Redundancy elimination: minimal set of actors.

sent in the vicinity of the event. For example, as shown in Fig. 6 if the minimal set of actors to cover the event area is 9 and if there are 20 actors in that region, then the remaining 11 actors need not act on the environment. In the same example, it might also be desired that only the minimal set of sensors sense and report the environment. This case corresponds to where both sensors and actors are a minimal set to cover the event. On the other hand, in some applications, unlike the above classifications (i.e., minimal and maximal usage of sensors and actors), there may be a need for specifying a certain redundancy level (i.e., the number of sensors/actors involved in sensing/acting is between the minimum and maximum value), as well.

Now, a stricter requirement in some applications might be that the regions covered by different actors are not only a minimal set but are also *mutually exclusive*. For example, if there is an application where the sensors report the amount of moisture in the ground and the actors have to irrigate the area uniformly, then the actors should not only cover the entire region but also make sure that the acting regions do not overlap.

For the type of transmission, there can be two possibilities such as single-hop and multi-hop. Although single-hop communication is always inefficient in WSNs due to the long distance between sensors and the sink, in WSANs this may not be the case, because actors are close to sensors as stated in Section 2. In fact, here the location of the actor determines the effectiveness of the single-hop communication. For example, if the event area is small and there is an actor in the middle of the event area, then the nodes located farther away from the actor have less energy burden. However, when the event area is large or the actors are at the edge or outside of the event area, multi-hop communication may be more efficient than single-hop communication due to the long distance between the actor node and the sensor nodes located farther away from the actor node. Therefore, the type of transmission depends on the deployment and location of actor nodes to which sensor data will be sent.

In addition to the type and requirements of the sensor-actor communication, as stated in the second problem above, there is a question in regard to which actor nodes will be informed about the event as a result of sensor-actor communication. In Section 3.1, we investigate how sensors can select actors to which they will send their data. In particular, we outline two cases called *Multi-Actor (MA)* and *Single-Actor (SA)*.

3.1. Actor selection

As shown in Fig. 7(a), in WSANs multiple actors can receive the information from sensors about the sensed phenomenon. We denote this case as *Multi-Actor (MA)* where every sensor

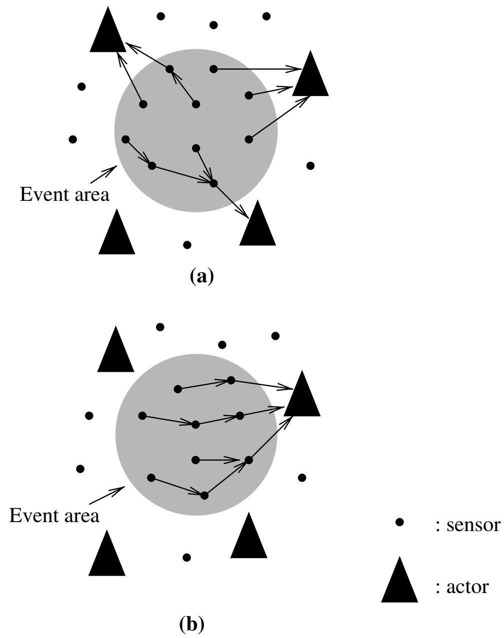

Fig. 7. (a) Multi-Actor (MA) vs. (b) Single-Actor (SA).

node can independently decide to which actor it will send its readings. However, the drawback of the lack of coordination between the sensors is that too many and unnecessary actors can be activated and as a result the total energy consumption of all sensors can become high.

To avoid this situation, sensors should coordinate with each other to form clusters. For each cluster there will be one actor to collect the data. These clusters may be formed such a way that:

- the event transmission time from sensors to actors is minimized, since low latency between sensing and acting is required in WSANs or
- the events from sensors to actors are transmitted through the minimum energy paths or
- the action regions of the actors can cover the entire event area.

Unlike in Fig. 7(a) where sensor readings are sent to multiple actors, only one actor receives event features, as shown in Fig. 7(b). We denote this case as *Single-Actor (SA)*. In fact, SA can be considered as a special case of MA. In SA, one of the main challenges is to determine the single actor node to which sensors will send their readings. Selecting an actor node may be based on some criteria such as:

- the distance between the event area and the actor should be small so that low delays are achieved and less power is consumed,
- the minimum energy path from sensors to the actor,
- the action range of the actor so that appropriate actions are performed on the event area.

Note that in the last case, there is no guarantee that the action range of the selected actor can cover the entire event area. Therefore, instead of considering distance, energy or timing issues, sensors may try to find the "best" actor for that event, i.e., the actor which has enough action coverage, energy and capability to perform the required action on the event area. In this situation, the actor receiving event information will be able to perform the required action itself without coordinating with other actors (see Section 4.1).

In SA, the actor can immediately perform the action if it has a wide action range and sufficient energy and also if this action can be performed only by a single actor. In the SA case, the latency between sensing and acting becomes low. However, if one actor is not sufficient for the required action or if it is not well-suited due to coverage and energy constraints to execute the action that actor publishes the *announcement message* (see Section 4) to other actors. Based on the feedback from other actors, it may select one or more other actors to perform the appropriate actions.

The advantage of MA over SA is that the MA provides actors to figure out where the center of an event is. The intensity of events may not be uniform inside an event area. As a result, the signal strengths from sensors to each actor may be different. Actors can compare their received signal strength values with each other and determine where the event intensity is dense. This can result in more effective actions by moving mobile actors towards the center of the event.

The disadvantage of MA is that actor–actor coordination is mostly based on the *negotiation* (see Section 4) among multiple actors unlike the *announcement message* in SA stated before. In MA, each actor may have some partial information about the overall event and thus, in order to take appropriate action decisions, actors must coordinate with each other as stated in Section 4, which may result in high communication overhead and high latency.

3.2. Research challenges

The following research issues related to SA and MA cases can be concluded for sensor-actor coordination in WSANs:

- For both SA and MA, in-sequence delivery of different events detected in a region may be required to ensure that there are no adverse effects on the target environment.
- In both SA and MA, it may be required to ensure synchronization in the reporting time of the sensed phenomena between different actors responsible for acting on the event.

- In certain applications where the events occur in different locations, for both SA and MA, it may be necessary that the sensed information is sent to an actor or to a set of actors determined based on the location of the event.
- In MA, it may sometimes be necessary to address the redundancy in the set of actors to which the sensed information is sent in order to save average energy consumed by the actors in the region. In these cases, it is necessary to send the information only to a subset of actors which cover the entire event region.
- As we have seen, there are trade-offs between SA and MA cases. The advantages and disadvantages of both cases need to be analytically investigated in order to figure out in which applications or situations which one is more efficient.

4. Effective coordination among actors

As stated in Section 2, in WSANs actors communicate with each other in addition to communicating with sensors. Actor–actor communication occur in the following situations:

- The actor receiving sensor data may not act on the event area due to small action range or insufficient energy.
- One actor may not be enough to perform the required action, thus, other nearby actors should be triggered.
- If multiple actors receive the same event information and there is an action threshold, these actors should "talk" to each other in order to decide which one of them should perform the action.
- In certain applications, if multiple actors are required to cover the entire event region, it may be necessary to ensure that these regions are non-overlapping or *mutually exclusive* in order to ensure uniform acting behavior over the entire region.
- If multiple actors receive information from multiple sensors for the same event, then it may be necessary to ensure that these multiple actors act on the environment at the same time. This *synchronization* requirement in the execution of the task is required in applications where a partial execution of the task alters the state of the event in the region where it has not been executed.
- In case of multiple events occurring simultaneously, task assignment can be done via actor–actor communication. Also, it may be desired that the tasks are executed sequentially. This constraint is referred to as *ordered execution* of tasks.
- After an actor node receives event information, if the event is spreading to other actors' acting areas, the actor node can transmit the sensor data or action command to those actors. In this way, there will be no need for sensors in those areas to send information to the nearby actors as they will be forwarded by initial set of actors. This is an alternative to the tracking problem identified in Section 3, where the actors handle different locations of the events.

All of the above situations which indicate the necessity of actor–actor coordination converge to the following question:

"Which actor(s) should execute which action(s)?"

The answer to this question can be given by exploiting the *coordination* between actor nodes. Actors should, whenever possible, coordinate strongly with each other in order to maximize their overall task performance [11]. Here, a task formally means an atomic unit of computation and control that actors will execute. However, in WSANs we call tasks as the necessary actions performed on the sensed events. Then, the above question can be restated as follows:

"How should multi-actor task assignment be done?".

The task assignment problems in WSANs can be examined by using the following two axes:

- *Single-Actor Task (SAT)* vs. *Multi-Actor Task (MAT)*: SAT means that each task requires exactly one actor, whereas MAT means that a task requires multiple actors. Thus, multi-actor task assignment problems involve tasks that require the combined effort of multiple actors.

- *Centralized Decision (CD)* vs. *Distributed Decision (DD)*: In WSANs there is a need to take a decision on the action to be performed according to the event. The decision can be performed in a centralized way (called CD) or in a distributed way (DD). DD allows neighboring actors to coordinate locally which provides timely actions and network size-independent coordination. On the other hand, CD provides action decisions to be taken in an organized way since the decision is taken only at one actor node which may be equipped with more powerful communication facilities.

In Section 4.1, we investigate how actors are assigned with a task by exploiting the above axes. In Section 4.2, we discuss the research challenges related to MAT, SAT, CD, DD cases.

4.1. Task assignment

In MAT, if multiple actors (MA) receive event information from sensors, in the DD case they *negotiate* with each other and coordinate locally to select the "best" [1] actors for the task. On the other hand, in the CD case, they directly transmit the specifications of the event such as location, intensity, etc. to the pre-determined actor node which functions as a decision center. This decision center which has already information about the actors in the network selects the "best" actors for that task and triggers them to initiate the action. These selected actors (both in DD and CD) may not be the ones which received sensor data via sensor-actor coordination, because actors receiving event information may not be the "best" actors for that task. For example, they may not be close enough to the event area, or they may not be capable of performing the required task.

In MAT, if only one actor (SA) receives event information at the end of sensor-actor coordination phase, there is still a need for coordination among actors in order to determine which actors will act on which part of the event area. However, in this case since all sensor data are collected at one actor that can function as the central decision unit. It then broadcasts an *announcement message* to other actors which contains the details about the event and the task. Based on the feedback from other actors, it selects "best" actors and assigns the action task to them.

After assigning action tasks, each selected actor initiates an action inside its action range [2]. However, in order to react to every data representing a phenomenon occurred inside a coverage area of the sensors, the union of the action ranges of selected actors should cover the entire *event area*.

Moreover, some parts in the event area may require more than one actor to perform an action depending on the event intensity and the capabilities of actors in that area.

On the other hand, if the total action range of the actors is much larger than it supposed to be, the actions may be performed outside the event area. Depending on the application, this may cause catastrophic results as well as unnecessary consumption of actor resources.

Similarly, when action ranges of actors intersect with each other and all actors act at the same time may cause catastrophic events (e.g., in the case of disposer of a tranquilizing gas actors).

Thus, while assigning tasks to actors, these action coverage challenges must also be taken into account.

As a result, regardless of the number of actors receiving sensor data, the objective of MAT is to select "best" actors while meeting action coverage requirements of the corresponding application and the event. The type of actor selection (i.e., MA or SA) in sensor-actor coordination phase only affects the coordination mechanism to select "best" actors.

Similarly, the main objective of SAT is to select the "best" actor (e.g., the one which is the "best" among all actors and which has the action range covering exactly the entire event area) for the action task. In fact, if the type of actor selection is

[1] Here, the "best" actor refers to the one which is close to the event area, or which has high capability and residual energy, or which has small action completion time at which it will complete its action.

[2] Depending on the characteristics of actors, if an actor is chosen to act, it either must act on the entire area in its action range or is able to act selectively on part of its action area.

MA, the coordination for SAT can be considered as a special case of MAT (i.e., actors receiving sensor data coordinate either in DD or in CD and select the "best" actor). However, if one actor is informed about the event features (SA), then there arises a question such that whether this actor takes a decision in an isolated fashion and thus initiates action by itself or it first communicates with other actors/decision center. Intuitively, if sensor-actor communication takes long time and the application is delay-intolerant (delay bound of received sensor data is low) as long as the actor can provide the minimum requirements of the task (e.g., it should be able to act on the whole event area and to have enough energy), initiating an action immediately is reasonable in order to perform the action on time. This way, maybe the action is not performed by the "best" actor, however, it is guaranteed that the action is completed in a timely manner. On the other hand, if the delay bound of the data is not very low or the actor does not provide the minimum requirements of the task, it should not immediately start to perform the action by itself, instead in CD it should communicate with the decision center and should allow it to choose the appropriate actor or in DD it should broadcast an *announcement message*, as explained before, to inform the other actors about the task and then should select the "best" one according to the responses from them.

4.2. Research challenges

As we have seen, actors coordinate explicitly and with purpose either in centralized way or in distributed way in order to solve the task assignment problems in WSANs. However, this coordination has the following challenges:

- Algorithms are needed which can provide actor(s) receiving event information to understand whether the task is a single-actor task (SAT) or a multi-actor task (MAT). For SAT case, the problem is how to select the single actor which will perform the action among all capable actors. For MAT case, the additional problem is how to decide on the optimum number of actors performing the actions.
- A communication model is needed between actors, which is valid for both SAT and MAT cases. Although, as mentioned in Section 2, actors can perform long-range communication and thus, generally can communicate directly with their neighbor actors, if the distance between neighbor actors is larger than the transmission range of actors, they cannot directly communicate with each other. In those situations, actors use sensor nodes as middlemen, which means that actor-actor coordination is performed via sensor nodes.
- In DD, for both SAT and MAT cases, in-sequence execution of different events detected in a region may be required to ensure that there are no adverse effects on the target environment. We refer to this requirement as the *ordered* execution of tasks for a series of events.
- In both DD and CD, for MAT, some applications may require *synchronization* of actors to act on the event at the same time. In this case, the actors have to coordinate either in a distributed or centralized fashion to determine the time of execution of the task.
- For both CD and DD and both SAT and MAT cases, when the events are in different locations, it may be necessary that the task is executed by a set of actors that are not necessarily close to the event location when it was first sensed. In these cases, based on the location of the event, the actors receiving the event information forward it to a different set of actors corresponding to the estimated new position of the event.
- In both CD and DD, for MAT, it may be necessary to address the redundancy in the set of actors that perform a task in order to save on the average energy consumed by the actors in the region. In these cases, it is necessary that only a subset of actors covering the entire event region is selected to carry out the task.
- In cases where the acting range is greater than the event region, for both SAT and MAT cases, it is necessary that the tasks are executed partially in that event region by one actor or by a set of actors. This *partial execution* of tasks requires some new ideas regarding the size of the event in the first place.

- In DD, for both SAT and MAT cases, depending on the number of actors receiving event information there occurs either *announcement message* (SA case) or *negotiation* (MA case). Therefore, there is a need to specify the contents of these messages, e.g., what are the fields of each message? Moreover, the algorithms which provide efficient data transmission for both types of messages need to be developed.
- In CD, the challenge is to select the actor which will function as a decision unit. Moreover, there is a need for effective mechanism which provides the decision unit actor to know the current characteristics (location, capability, etc.) of other actors in the network so that it can trigger the most appropriate actors for the task.
- One of the most important requirements of actor-actor coordination is to minimize the task completion time. Thus, coordination and communication protocols should support real-time properties of WSANs.

5. Protocol stack for WSANs

To date there exists no standardized protocol stack for WSNs and WSANs. We suggest that the protocol stack for sensor and actor nodes may basically consist of three planes, (i.e., communication plane, coordination plane, and management plane) shown in Fig. 8. Communication plane [3] enables the information exchange among the nodes of the network. Data received by a node at the communication plane are submitted to the coordination plane which decides how the node acts on the received data. Moreover, the coordination plane provides nodes to be modelled as a social entity, i.e., in terms of the coordination and negotiation techniques it possesses. Management plane is responsible for monitoring and controlling a sensor/actor node so that it operates properly. It also provides information needed by the coordination layer to make decisions.

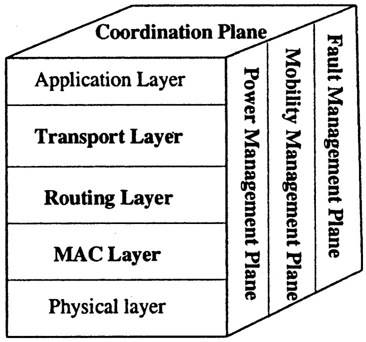

Fig. 8. WSAN protocol stack.

In the following three subsections, we discuss the requirements and characteristics of each plane for both sensor-actor and actor–actor coordinations.

5.1. Management plane

The functions performed by the management layer can be categorized into the following three areas:

- *Power management plane* manages how a node uses its power. For example, when the power level of a node is low, this plane informs the coordination plane so that the node will not participate in sensing, relaying, or acting activities.
- *Mobility management plane* detects and registers the movements of nodes so that network connectivity is always maintained.
- *Fault management plane* refers to the detection and resolution of node problems. For example, when the sensitivity of sensing unit or the accuracy of the actuation unit degrades, fault management plane informs the coordination plane about this situation.

5.2. Coordination plane

Coordination plane determines how a node behaves according to the data received from commu-

[3] Communication plane consists of five subplanes (i.e., layers), namely application, transport, routing, MAC, and physical layers.

nication plane and management plane. After sensing an event, sensors communicate their readings with each other. At each sensor node these exchanged data are submitted to the coordination plane to make decisions. This way, sensors are able to coordinate among themselves on a higher-level sensing task. Moreover, sensor–sensor coordination may also be required to determine nodes which will not transmit data (due to low power or applied MAC protocol), to perform multi-hop routing and data aggregation and the most importantly to select actor(s) to which sensor data will be transmitted.

The existence of coordination plane may be much more critical for actors than for sensors, since actors may need to collaborate with each other in order to perform appropriate actions. When an event occurs, the common goal of all actors is to provide required action on that event. Thus, social abilities, i.e., sophisticated coordination and negotiation abilities, are necessary in WSANs to ensure coherent behavior in the community of actors. These required social abilities of an actor are defined in the coordination plane. Specifically, which layer in actor–actor coordination is responsible to make decisions about which actors act on which part of the event area and whether to have these actors act concurrently or, if sequentially, then in what order [17].

5.3. Communication plane

Communication plane receives commands from coordination plane (about the decision of how the node will behave) and according to that information provides the link relation between nodes by using communication protocols. Specifically, the communication plane deals with the construction of physical channels, the access of the node into the medium (MAC), the selection of routing paths through which the node transmits its data and the transport of packets from one node to another.

In the following subsections, we investigate the requirements and challenges of the transport, MAC and routing layers as well as the cross layer integration between these layers both for sensor-actor and actor–actor communications.

5.3.1. Transport layer

In addition to the *conventional reliability* the new transport protocols must also support real-time requirements in WSANs. Several transport layer protocols have been developed for ad-hoc networks and wireless sensor networks in recent years [2,5,12]. However, to the best of our knowledge there exist no transport protocols which deal with both the reliability and real-time for WSANs to date. For instance, when the transport protocol for sensor-actor communication detects low reliability, transport protocol for actor–actor communication regulates the traffic between actors so that the actor receiving low reliable event information can inform the other nearby actors about this situation as soon as possible. Since sensor-actor and actor–actor communications occur consecutively in WSANs, a unified transport protocol is needed which works well for both cases.

5.3.2. Routing layer

In WSANs, when sensors detect an event, there is no specific actor to which a message will be sent. This uncertainty occurring due to the existence of multiple actors causes challenges in terms of routing solutions.

First selecting an actor node is one of the challenges for a source sensor node. The source data should then be routed towards the selected actor in an energy efficient way. While the source data is transmitted through relaying sensors towards to an actor node, it may be aggregated or forwarded in order to achieve high efficiency. In addition to determining the path selection and data delivery, routing protocol should support real-time communication by considering different deadlines due to different validity intervals. Moreover, the routing protocol should also consider the issue of prioritization and should provide data with low delay bounds to reach the actor on time.

In recent years there has been a considerable amount of research on routing problems in sensor networks [1].

An anycast mechanism developed in [15] does not support the sensor-sensor coordination occurring in WSANs due to the result of correlated information among multiple sensor sources which detect the same event. Moreover, this mechanism causes a sensor which is one hop away from an actor to receive also interests from an actor on the other side of the network. This may cause unnecessary traffic load in the network.

SEAD developed in [16] is also not suitable for WSANs since it does not deal with end-to-end delay minimization which is one of the main goals in WSANs. Furthermore, it is developed for the case where all sinks request data from one source at refresh rates, whereas in WSANs only actors which are in the vicinity of a phenomenon are interested in the event information.

SPEED [14] is an adaptive, location-based real-time routing protocol which can be effectively used if the location information is available in all sensor nodes and the location updates can be delivered to the source sensors regularly. However, SPEED is not suitable for WSANs since it does not support Multi-Actor (MA) case and the mobility of actors.

Moreover, resource-limited sensor nodes and higher energy capacity cluster heads are used in [18]. This model may be suitable for WSANs such as an actor can become a cluster head and each source sensor can become a member of a cluster.

However, several open research issues must be investigated such as

- How are the clusters formed, e.g., are they formed based on the event?
- How will the clusters be adaptive to mobility, or
- How will the clusters satisfy the real-time constraints?

For actor–actor communication, routing protocols developed for ad-hoc networks such as DSR, AODV, OLSR [7] can be used as long as they are improved so that real-time requirements are met and communication overhead occurring at sensor nodes due to actor–actor communication is low.

5.3.3. Medium access control

In order to effectively transmit the event information from large number of sensors to actors there is a need for MAC protocol. Moreover, in some applications, (i.e., distributed robotics) actors may be mobile. As they move, they may leave the transmission regions of some sensors and enter other sensors' region or they may become totaly disconnected from the network. Therefore, another function of MAC protocol in WSANs is to maintain network connectivity between sensors and mobile actors. Furthermore, as discussed before, the timely detection, processing, and delivery of information are indispensable requirements in a sensor/actor network application.

Classical contention-based protocols are not appropriate for real-time sensor-actor communication since contention-based channel access requires handshaking which increases the latency of the data. TRACE [23] is a reservation TDMA protocol which suffers from the added overhead for reservation contention while PBP (Predictive Backoff Protocol for IEEE 802.11) suffers from the requirement of large amount of energy due to all sensors listening to others' transmissions.

By exploiting the periodic nature of the sensor network traffic, a collision-free real-time scheduling algorithm is presented in [3]. Collision-free protocols may be suitable for WSANs, because they can potentially reduce the delay and provide real-time guarantees as well as save power by eliminating collisions. A problem in a large class of current collision-free protocols is the use of multiple channels [3]. This imposes a nontrivial requirement on the hardware of the nodes in the network as mentioned in [21]. Thus, further study is needed to tell whether the performance gain would overcome the increased cost of the hardware. Moreover, in [3] and generally in all existing collision-free protocols the mobility is not investigated.

For actor–actor communication, the existing MAC protocols developed for ad-hoc networks cannot be directly used. They should be improved so that they support real-time traffic, since in WSANs, depending on the application, interaction with the world may impose a real-time constraint on computation and communication.

5.3.4. Cross-layering

Current WSN and WSAN protocol designs are largely based on a layered approach. However, the suboptimality and inflexibility of this paradigm result in poor performance for WSANs, due to constraints of low energy consumption and low latency. Therefore, instead of having individual layers, we may need cross-layering where layers are integrated with each other.

In WSANs, one of the main factors which causes low *event reliability* is network congestion. In the case of high congestion, MAC layer reacts locally by exponential back-off [7], while transport layer reacts by lowering the transmission rates of sensors. However, normally these two layers act independently from each other which causes inefficiencies due to the duplication of functions. By cross-layering approach, each protocol shares its data with other protocols, which avoids those inefficiencies. For example, in WSANs when congestion is high, first of all MAC layer reacts to the congestion. If this response is not sufficient, MAC layer informs the routing layer about this congestion. Then, routing layer lets coordination plane know the situation. As a result, coordination plane and routing layer provide data traffic to be rerouted through another appropriate actor node. On the other hand, if alternate actors and routes do not exist, transport protocol mechanisms can be used to freeze traffic transmissions.

Another example of the cross-layering design in WSANs is the optimization of the size of the packets transmitted from sensors to actors. In order to provide a unified packet structure that incorporates the functionalities of each protocol in the protocol stack, routing, MAC and physical layers should be investigated together. The energy efficiency of the WSAN depends on the energy required to transmit a packet and the reliability of the network. From the routing layer point of view, reliability of the packet depends on the distance of the node generating the packet in terms of the number of hops to the actor. Intuitively, it is better to send smaller sized packets from the nodes far away from the actor. Hence, in order to provide energy efficiency, the information about an event may be transmitted to the actor using small sized packets while the relay nodes aggregate the packets due to being closer to the actor.

On the other hand, the size of the packet determines the number of packets needed to be sent to inform an event to the actor. Then, from the MAC layer point of view, the number of packets translates into the number of contention attempts the node needs to perform. Decreased packet size in effect leads to increased collision probability and thus, high energy consumption at the MAC layer. Lastly, from the physical layer point of view, as the coding rate increases, communication will be more reliable. Increased rate translates into sending more bits for useful information. However, a sensor node consumes energy based on the number of bits it sends for a transmission, i.e., packet size. Hence, packet size optimization also affects the bit level energy consumption. As a result, a useful model and an energy efficiency metric that accommodates all these factors is needed for optimization of packet sizes in WSANs.

In addition to the interactions among transport, MAC, routing and physical layers, in WSANs, there should also be interdependency between the application layer and those lower layers. Application layer must adapt to time-varying QoS parameters offered by the lower layers. While the network provides the best possible QoS to the application, this QoS will vary with time as channel conditions and network topology change. Thus, applications must also adapt to the QoS offered.

The basic ideas of cross-layering optimization stated above are also valid for actor–actor communication. However, since in WSANs, as mentioned in Section 1, actors may be mobile, link characteristics and network topology may change rapidly. Then, in case of the weak link connectivity between actors, physical layer at which link connectivity can be measured accurately and quickly responds to this situation by increasing its transmit power or its error correction coding. However, if the weak link is caused by something difficult to correct at the physical layer, i.e., high mobility of nodes, it is better for the physical layer to interact with the higher layers [12]. For example, in WSANs actors may perform unicast communication instead of broadcasting in order to prevent resource-constrained sensors from receiving unnecessary messages. However, in case of the high mobility, informing the routing layer might change the routing strategy from unicast to broadcast in the general direction of the intended actor. Hence, in the cross-layering approach, each layer of the protocol stack not only responds to local variations, but also responds to the information from other layers [5,12].

6. Conclusion

The effectiveness of the sensor networking can experience a profound leap if the actors are also an integral part of the deployed network. When the sensor field is complemented with actors, there will be one more option called acting as well as sensing and deciding for human controller. On the other hand, realization of wireless sensor and actor networks (WSANs) needs to satisfy the requirements introduced by the coexistence of sensors and actors. Throughout this paper, we explain the research challenges occurring due to sensor-actor and actor–actor coordinations in WSANs and investigate how the communication protocols in WSANs will be different from the protocols in WSNs.

However, there are several open research issues that should be investigated in WSANs:

- For sensor-actor coordination, algorithms that can provide ordering, synchronization and eliminate the redundancy of actions need to developed.
- For actor–actor coordination, there is a need to provide a unified framework that can be exploited by different applications to always select the best networking paradigm available according to the events sensed and to the operation to be performed, so as to provide efficient actor–actor communication.
- There is a need for an analytic framework in order to characterize the three planes, that is, management, coordination and communication planes stated in Section 5.
- As mentioned in Section 5.2, sophisticated distributed coordination algorithms need to be developed for effective sensing and acting tasks.
- As stated in Section 5.3, in WSANs the application, transport, routing, MAC and physical layers have common requirements and are highly dependent on each other. Hence, lever-aging a cross layer approach can provide much more effective sensing, data transmission, and acting in WSANs. Several cross-layer integration issues among the communication layers should be investigated in order to improve the overall efficiency of WSANs.
- Finally, maybe the most importantly for some applications there is a need for real-time communication protocols for both sensor-actor and actor–actor coordinations in WSANs.

Acknowledgment

We thank Ozgur Akan and Eylem Ekici for their feedback and suggestions.

References

[1] K. Akkaya, M. Younis, A survey on routing protocols for wireless sensor networks, AdHoc Networks, in press, 10.1016/j.adhoc.2003.09.010.
[2] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: A survey, Computer Networks 38 (4) (2002) 393–422.
[3] M. Caccamo, L.Y. Zhang, L. Sha, G. Buttazzo, An implicit prioritized access protocol for wireless sensor networks, in: Proc. IEEE Real-Time Systems Symp., December 2002, pp. 39–48.
[4] T.W. Carley, M.A. Ba, R. Barua, D.B. Stewart, Contention-free periodic message scheduler medium access control in wireless sensor/actuator networks, in: Proc. of Real-Time Systems Symposium, Cancun, Mexico, December 2003.
[5] I. Chlamtac, M. Conti, J.N. Liu, Mobile ad-hoc networking: imperatives and challenges, Ad Hoc Networks 1 (1) (2003) 13–64.
[6] M. Chow, Y. Tipsuwan, Network-based control systems, in: Proc. of IEEE IECon 2001 Tutorial, Denver, CO, November 28–December 2, 2001, pp. 1593–1602.
[7] M. Conti, S. Giordano, G. Maselli, G. Turi, Cross-layering in mobile ad-hoc network design, IEEE Computer, Special Issue on AdHoc Networks 37 (2) (2004) 48–51.
[8] DARPA Tactical Mobile Robotics, http://www.darpa.mil/ato/programs/tmr.htm, visited in March 2004.
[9] H.R. Everett, D.W. Gage, A Third Generation Security Robot, SPIE Mobile Robot and Automated Vehicle Control Systems, Vol. 2903, Boston, MA, November 1996.
[10] B.P. Gerkey, M.J. Mataric', A market-based formulation of sensor-actuator network coordination, in: Proc. of the AAAI Spring Symposium on Intelligent Embedded and Distributed Systems, Palo Alto, CA, March 25–27, 2002, pp. 21–26.
[11] B.P. Gerkey, M.J. Mataric, Sold!: Auction methods for multi-robot coordination, IEEE Transactions on Robotics and Automation, Special Issue on Multi-robot Systems 18 (5) (2002) 758–768.
[12] A.J. Goldsmith, S. Wicker, Design challenges for energy-contrained ad-hoc wireless networks, IEEE Wireless Communications 9 (4) (2002) 8–27.

[13] M. Haenggi, Mobile sensor-actuator networks: opportunities and challenges, in: Proc. 7th IEEE Int. Workshop, Frankfurt, Germany, July 2002, pp. 283–290.
[14] T. He, J. Stankovic, C. Lu, T. Abdelzaher, SPEED: A real-time routing protocol for sensor networks, in: Proc. IEEE Int. Conf. on Distributed Computing Systems (ICDCS), Rhode Island, USA, May 2003, pp. 46–55.
[15] W. Hu, N. Bulusu, S. Jha, An anycast service for hybrid sensor/actuator networks, in: Proc. of the 15th IEEE Int. Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Barcelona, Spain, 5–8 September 2004.
[16] H.S. Kim, T.F. Abdelzaher, W.H. Kwon, Minimum-energy asynchronous dissemination to mobile sinks in wireless sensor networks, in: Proc. of the First ACM Int. Conf. on Embedded Networked Sensor Systems (ACM Sensys'03), November 2003, pp. 193–204.
[17] V. Lesser, Reflections on the nature of multi-agent coordination and its implications for an agent architecture, In Autonomous Agents and Multi-Agent Systems 1 (1) (1998) 89–111.
[18] V. Mhatre, C. Rosenberg, D. Kofman, R. Mazumdar, N. Shroff, A minimum cost surveillance sensor network with a lifetime constraint, IEEE Transactions on Mobile Computing, in press.
[19] E.M. Petriu, N.D. Georganas, D.C. Petriu, D. Makrakis, V.Z. Groza, Sensor-based information appliances, IEEE Instrumentation and Measurement Magazine 3 (4) (2000) 31–35.
[20] Sandia National Laboratories, http://www.sandia.gov/media/NewsRel/NR2001/minirobot.htm, visited in March 2004.
[21] J.A. Stankovic, T.F. Abdelzaher, C. Lu, L. Sha, J. Hou, Real-time communication and coordination in embedded sensor networks, Proceedings of the IEEE 91 (7) (2003) 1002–1022.
[22] Sub-kilogram intelligent tele-robots (SKITs), http://www-robotics.usc.edu/~behar/SKIT.html, visited in March 2004.
[23] B. Tavli, W. Heinzelman, TRACE: Time reservation using adaptive control for energy efficiency, IEEE Journal on Selected Areas of Communication 21 (10) (2003) 1506–1515.
[24] S. Thrun, M. Diel, D. Hähnel, Scan alignment and 3-D surface modeling with a helicopter platform, in: Proc. of the Int. Conf. on Field and Service Robotic (FSR'03), July 2003.

Ian F. Akyildiz received his BS, MS, and PhD degrees in Computer Engineering from the University of Erlangen-Nuernberg, Germany, in 1978, 1981 and 1984, respectively.

Currently, he is the Ken Byers Distinguished Chair Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology and Director of Broadband and Wireless Networking Laboratory.

He is an Editor-in-Chief of Computer Networks (Elsevier) and for the newly launched Ad-Hoc Networks (Elsevier) Journal.

He is an IEEE FELLOW (1995), an ACM FELLOW (1996). He served as a National Lecturer for ACM from 1989 until 1998 and received the ACM Outstanding Distinguished Lecturer Award for 1994.

He received the 1997 IEEE Leonard G. Abraham Prize award (IEEE Communications Society) for his paper entitled "Multimedia Group Synchronization Protocols for Integrated Services Architectures" published in the IEEE Journal of Selected Areas in Communications (JSAC) in January 1996.

He received the 2002 IEEE Harry M. Goode Memorial award (IEEE Computer Society) with the citation "for significant and pioneering contributions to advanced architectures and protocols for wireless and satellite networking".

He received the 2003 IEEE Best Tutorial Award (IEEE Communication Society) for his paper entitled "A Survey on Sensor Networks", published in IEEE Communication Magazine, in August 2002.

He received the 2003 ACM SIGMOBILE award for his significant contributions to mobile computing and wireless networking.

His current research interests are in Sensor Networks, InterPlaNetary Internet, Wireless Networks and Satellite Networks.

Ismail H. Kasimoglu received his B.Sc. degree in Electrical and Electronics Engineering from Bilkent University, Ankara, Turkey, in 2003. He is currently a Research Assistant in the Broadband and Wireless Networking Laboratory and pursuing his Ph.D. degree at the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA. His current research interests include wireless sensor and actor networks.