

Research Article

Adaptive Learning Based Scheduling in Multichannel Protocol for Energy-Efficient Data-Gathering Wireless Sensor Networks

Kieu-Ha Phung,^{1,2} Bart Lemmens,^{1,3} Mihail Mihaylov,⁴ Lan Tran,² and Kris Steenhaut^{1,3}

¹ Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel, Pleinlaan 2, 1050 Brussel, Belgium

² School of Electronics and Telecommunications, Hanoi University of Science and Technology, Dai Co Viet 01, Hanoi, Vietnam

³ Department IWT, Erasmushogeschool Brussel, Nijverheidskaai 170, 1070 Brussel, Belgium

⁴ Faculty of Science and Bio-Engineering Sciences (DINF), Vrije Universiteit Brussel, Pleinlaan 02, 1050 Brussel, Belgium

Correspondence should be addressed to Kieu-Ha Phung; kphung@etro.vub.ac.be

Received 26 October 2012; Accepted 8 January 2013

Academic Editor: Jiman Hong

Copyright © 2013 Kieu-Ha Phung et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multichannel communication protocols have been developed to alleviate the effects of interference and consequently improve the network performance in wireless sensor networks requiring high bandwidth. In this paper, we propose a contention-free multichannel protocol to maximize network throughput while ensuring energy-efficient operation. Arguing that routing decisions influence to a large extent the network throughput, we formulate route selection and transmission scheduling as a joint problem and propose a Reinforcement Learning based scheduling algorithm to solve it in a distributed manner. The results of extensive simulation experiments show that the proposed solution not only provides a collision-free transmission schedule but also minimizes energy waste, which makes it appropriate for energy-constrained wireless sensor networks.

1. Introduction

Recently envisioned applications of wireless sensor networks (WSNs) exhibit more stringent Quality of Service demands in parallel with the traditional lifetime optimization problem. For example, Wireless Multimedia Sensor Networks (WMSNs), which enable new surveillance, traffic monitoring, and healthcare systems [1], clearly require large bandwidth. As the hardware costs are continuously decreasing, the number of WSN deployments is expected to grow drastically. Network density will increase greatly, resulting in high levels of inter-WSN interference. A new challenge needs to be addressed: the increased bandwidth demand in the presence of higher levels of interference.

The challenges stem from the fact that the bounded wireless channel capacity needs to be shared between nodes within the same geographical area. Meanwhile, most sensor nodes available on the market are equipped with IEEE 802.15.4 compliant transceivers which are able to operate on 16 nonoverlapping channels in the 2.4 GHz ISM band. Thus, using the available multiple channels efficiently to exploit parallel transmissions in WSNs becomes very attractive.

Although multichannel communication is quite well-investigated in wireless network [2–4], most proposals are not directly applicable to WSNs. Firstly, a sensor node is a low-power device with a single interface transceiver incapable to operate on several channels simultaneously, unlike other wireless devices such as laptop computers or personal digital assistants. As a consequence, a transmission will fail if the sender and its intended receiver are not on the same channel, which is called the deafness problem in multichannel communication. Secondly, dynamic channel negotiation or RTS/CTS (request to send/clear to send) exchange used in some approaches [5, 6] causes large overhead, as the size of WSN packets is much smaller (30 Bytes) than the packet size of the IEEE 802.11 standard (512 Bytes). Last, the bandwidth of sensor radios is limited compared to other wireless devices, for example, 50 Kbps to a maximum value of 250 Kbps.

Recently, several multichannel protocols have been designed specifically for WSNs. Fixed channel assignment approaches often divide the network into clusters or subtrees [7, 8] to which different channels are assigned to alleviate interference among clusters or subtrees. Those approaches cannot reduce interference efficiently because they do not

fully exploit the routing topology which has a large influence on the interference in the network. In [9], Yu et al. propose a game theoretic based channel assignment algorithm in which they take into account the routing information to reduce the interference more effectively but they do not fully achieve interference-free communication. Some other approaches schedule transmissions both in the time and frequency domain to achieve collision-free access [10, 11]. The scheduling algorithm proposed in [10] operates in a centralized manner, which causes extra overhead to communicate the assignment to each node. Moreover, they do not support heterogeneous traffic patterns.

A common application of WSNs is data gathering, where the many-to-one/converge-cast communication mode is usually taking place. In a multihop scenario, sensed data from nodes, called *source nodes*, are all relayed through one or several intermediate nodes, towards one or several base stations, called *sinks*. Therefore, the routing topology is often in the form of a tree or a forest, in which a node has one or several parent nodes. In fact, it is difficult to fully exploit routing information to achieve a collision-free channel assignment and transmission schedule since nodes must share information dynamically and be involved in the channel assignment or scheduling process. In such cases, the problem becomes NP-hard [9, 10].

Reinforcement Learning (RL) opens up a new path for practical WSN approaches. RL has already been applied to several problems in clustering, routing and neighbourhood management, medium access control, and radio duty cycling [12, 13]. It is a decentralized technique of goal-directed learning by trial and error [14]. Every node behaves as an autonomous agent, acting independently, obtaining the feedback from its environment, and learning as a result of this feedback. The feedback is in fact the result of the actions of all nodes in the system. Based on the obtained feedback, the agent adjusts its action in pursuit of its goal set by the system designer. In this paper, we aim at maximizing parallel transmissions to improve the network throughput by scheduling nodes' transmissions in order to avoid the collisions and the deafness problem in multichannel operation for data gathering WSNs. The elimination of the collisions and the deafness problem minimizes the energy consumption at each node. We show how sensor nodes learn to achieve successful transmissions and receptions in a decentralized manner, while minimizing energy consumption.

In this paper, we propose a multichannel protocol for data gathering WSNs with a Reinforcement Learning based scheduling algorithm. The proposed protocol exploits routing information for scheduling collision-free transmissions over multiple channels. The algorithm operates in a distributed manner and especially reduces the energy wasted by collision, idle listening, and deafness problem. Its performance has been evaluated through extensive simulations.

The rest of the paper is organized as follows: in Section 2, we present the background of the multichannel problem in energy-constrained WSNs and the formulation of the joint routing and scheduling problem in Section 3. The RL based scheduling algorithm and the details of the multichannel protocol are presented in Sections 4 and 5, respectively. In

Section 6, the results of extensive experiments are presented to assess the performance of the proposed protocol. A discussion on the advantages and disadvantages of the proposed protocol is given in Section 7. Section 8 concludes this paper.

2. Preliminaries and Problem Formulation

2.1. Network Throughput and Multichannel Operation. In data gathering wireless sensor networks, the overall bound on the average network throughput per node is W/n , in which n is the number of source nodes and W is the transmission capacity [15]. Note that source and sink nodes are all equipped with half-duplex transceivers. However, it is proved that the maximum throughput can be reached only if the sink is 100% busy receiving packets and if the schedules of all nodes are aligned for interference-free communication for the given network topology. The main reason for the limitation on network throughput is *the interference in the shared wireless medium*. As depicted in Figure 1(a), while node 1 is transmitting to the sink, node 2 must be inactive and nodes 3, 4 can be inactive or communicate with each other on a different channel. Otherwise, they will create collision to the transmission between node 1 and the sink since they are all in the same collision domain.

Enabling interference-free spatial reuse of the shared wireless medium to improve network throughput has been an important research topic in the wireless networks domain [16]. There exist many different medium access control (MAC) methods designed for coordinating communication. The main techniques used to share the wireless medium and alleviate conflicts are contention based (carrier sense multiple access (CSMA)) or schedule based, such as time division multiple access (TDMA), frequency division multiple access (FDMA), and code division multiple access (CDMA). In addition, some approaches use power control and directional antennas to further reduce interference.

Figure 1(b) illustrates a schedule of TDMA collision-free transmissions for all source nodes to forward their data to the sink using a single channel/frequency for the line topology network in Figure 1(a). They are all in the same collision domain hence only one transmission can happen at one moment and the maximum achievable throughput is $W/5$. Higher network throughput can be achieved using multiple channels, depicted in Figure 1(c) (two channels in this case). Multichannel usage allows parallel transmissions within a single collision domain, which results in increased throughput, decreased collision probability, and thus better energy efficiency.

However, as the half-duplex transceivers of the sensor nodes are only capable of operating on a single channel at any time, a transmitter and a receiver must tune their radios to the same channel to communicate successfully. Given the same geographical network topology but different routing trees, the maximum achievable network throughput can differ as illustrated in Figure 2. When node 1 is chosen as the common parent of node 3 and node 4 for relaying their data, the maximum achievable network throughput is $W/5$, since node 1 cannot receive simultaneously from nodes

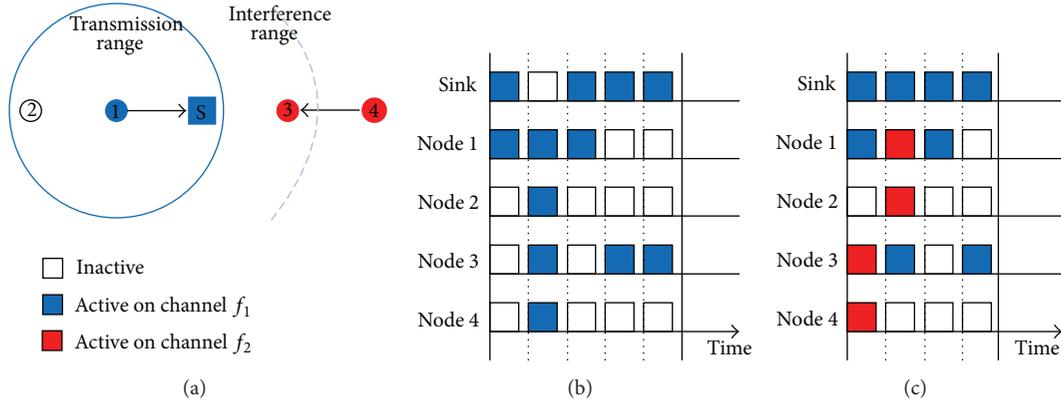


FIGURE 1: Network throughput, and collision-free transmission schedule in line topology. (a) Node 1 is transmitting to the sink, node 2 must be inactive on the same channel, and nodes 3, 4 are inactive or communicating on different channels. (The arrows point from sender to receiver.) (b) Time schedule of transmissions on single channel network. Each line depicts the operation of nodes. The achievable throughput is $W/5$ since it needs 5 time slots to transfer all data to the sink. (c) Time and frequency schedule of transmissions with two channels (the achievable throughput is $W/4$).

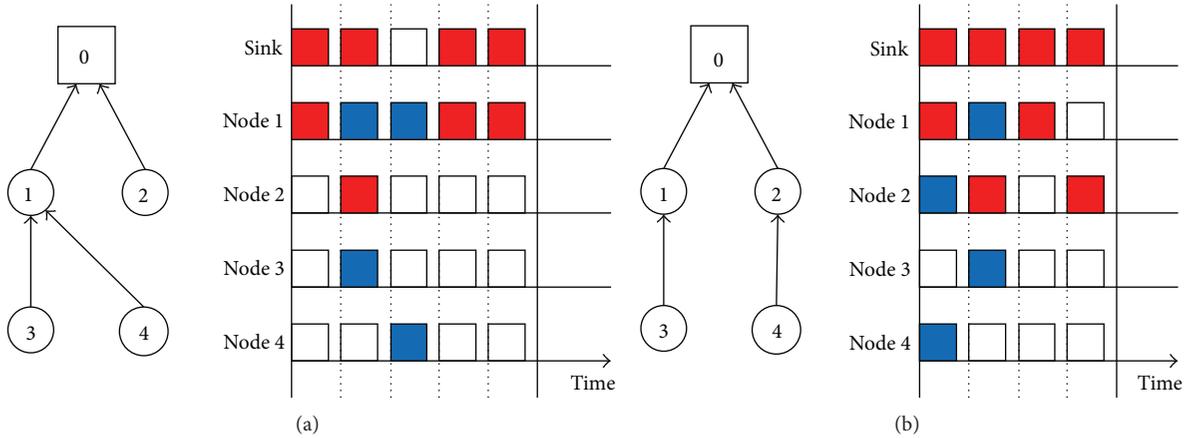


FIGURE 2: The influence of routing topology on achievable network throughput in the presence of half-duplex transceivers. In (a) a fixed routing structure is depicted in which nodes 3 and 4 have node 1 as relaying node. In this case, multichannel operation cannot increase throughput since node 1 can only receive or transmit on a single channel at any time. When nodes 3 and 4 select a different relaying node, as depicted in (b) the transmissions can be scheduled to maximize the number of concurrent parallel transmissions and the achievable throughput is $W/4$. (The arrows point from sender to receiver.)

3 and 4. In order to maximize the number of concurrent parallel transmissions, nodes 3 and 4 should forward their data to nodes 1 and 2, respectively. Therefore, it is possible to schedule node 1 transmitting to the sink while node 2 receiving packet from node 4 as depicted in Figure 2(b). It results in a maximum achievable network throughput of $W/4$. This proves that the routing topology can be managed for maximizing throughput.

2.2. Energy Efficiency and Channel Coordination. As can be seen in Figures 1 and 2, achieving a successful transmission between a pair of nodes in a multichannel wireless network requires the coordination between the sender and the receiver to transmit and to listen, respectively, at the same moment on the same channel. Furthermore, at that moment all other

nodes in range must refrain from transmitting on that channel to avoid interference with the ongoing transmission. This coordination of transmissions provides opportunities to save energy at both the sender (no retransmissions due to *collision*, elimination of *hidden-terminal* and *deafness problem*) and at the receiver (no *idle listening* and *overhearing*).

The deafness problem occurs when a transmitter sends a control packet (e.g., request to send RTS) to initiate a transmission and the transceiver of the destination is tuned to another channel. After sending several requests, if the transmitter did not get any response (e.g., clear to send CTS) it may conclude that the receiver is not reachable anymore. Besides being confronted with the hidden-terminal problem as known in single channel, a multichannel operated network has to deal with the effect of the inability to receive control packets sent on a different channel than the one the node's

transceiver is currently tuned to. This phenomenon is called the multichannel hidden-terminal problem [3]. The nodes that miss the control packets might thus falsely conclude the channel is idle and start transmitting on that channel which will cause a collision. The overhearing of messages happens when a node receives messages that are destined to other nodes. The energy spent for receiving a message can even be a bit higher than the energy spent for transmitting a message [17]. These problems are the main sources of energy wasted in communication, which is the major contribution of energy consumption in WSNs, and shorten the lifetime of the network significantly.

The above-mentioned problems require the coordination of the transmissions between nodes. Several approaches have been proposed and [16] provides a detailed survey. For example, the dedicated control channel approach, a subset of dynamic channel assignment, uses a channel exclusively for control purposes [18]. On this control channel, nodes negotiate which channel to use for the actual data transmission. The dedicated control channel acts, however, as a bottleneck and results in a maximum achievable network throughput equal to the one of a single channel network. The frequency-hopping approach used in [5, 6] improves the network throughput by allowing parallel rendezvous transmissions without channel negotiation or information exchange. However, it encounters difficulties in dealing with the deafness and multichannel hidden-terminal problems. The well-known (Carrier Sense Multiple Access with Collision Avoidance) CSMA/CA handshaking process, RTS/CTS exchange, used in [5] cannot solve these problems completely. An RTS/CTS scheme is furthermore not suitable for WSNs due to the small packet size of IEEE 802.15.4, which would impose a large control overhead.

3. Model for Joint Routing and Transmission Scheduling

We aim to design a multichannel protocol that not only addresses the problems of collision, idle listening, overhearing, deafness, and hidden terminal, but also maximizes the network throughput. Taking into account the effect of the routing topology on the number of parallel transmissions, a node should select simultaneously to which parent to forward and on which channel to transmit. The problem becomes a *joint routing and transmission scheduling* problem.

The proposed protocol is based on a combination of TDMA and FDMA techniques. TDMA is known to provide collision-free operation and excellent energy efficiency due to the minimization of idle listening and overhearing. However, a practical implementation of TDMA in sensor networks is not trivial due to time synchronization requirements and scalability issues which will be discussed in Section 7.

Time is discretized into fixed length frames composed of a number of time slots. The length of a slot allows the transmission of a single data message and an acknowledgement message. Instead of using a fixed frequency assignment for each node, a channel-hopping scheme is used as follows.

- (i) In each frame, a node periodically switches its channel at each time slot according to a chosen channel-hopping pattern, called the *default sequence*. A pseudo-random number generator is used to generate this sequence, in which the address of the node serves as the seed. In the sequence, there can be several broadcast slots in which all nodes hop to the same frequency. They can be used for local broadcast communication required by many upper-layer applications, for example, for updating a route metric.
- (ii) In order to establish communication, the sender tunes its radio to the receiver's current channel and transmits the data. The sender can reproduce the listening channel of the receiver when it knows the receiver address. Therefore, it allows parallel transmissions between several pairs of nodes without exchanging information or negotiating a communication channel which pose a major challenge for energy and bandwidth constrained WSNs.

In each time slot, a node can be in three states, listening on its default channel, deviating to another channel to transmit data, or being radio-off to save energy resource.

We consider a multihop data-gathering network comprising a single base station and many sensor nodes sensing/monitoring data (e.g., environment parameters) which has to be forwarded to the base station over a routing tree. Each node has a nonempty set of parent nodes; otherwise, the node is disconnected from the network. The generated traffic at each node is assumed to be periodic with respect to the frame length, but the pattern may be different. Thus, heterogeneous traffic loads within the network are allowed. A collision occurs at a receiver when more than one node in the receiver's collision domain is transmitting on the channel the receiver is listening on. It also includes the hidden-terminal problem. For reliable communication, a transmission is considered successful if the sender receives the acknowledgement from the intended receiver.

Since a node normally has several parent nodes (each hopping according to their respective default frequency sequence), a node might have several channels to switch to. As illustrated in Figure 2, a fixed routing structure might prevent an optimal exploitation of the multiple channels. This problem can be addressed by letting the nodes decide which parent they should forward their data to depending on which channel the parent is listening. The selection of the parent hence may be different for each time slot.

During a frame, each node has a "pool" of actions it can perform. A node should specify the exact action, stay on its home channel, or deviate to a chosen channel to transmit in a certain time slot of the frame or withdraw from communication.

It is clear that without coordination, the transmission can fail because of collision, deafness problem, or (multichannel) hidden-terminal problem, see Section 2.2. The objective of the schedule algorithm is to coordinate the action of each node in each time slot to avoid as much as possible failed transmissions while forwarding all the generated data toward the sink.

4. Reinforcement Learning Based Scheduling Algorithm

The proposed scheduling algorithm is based on Reinforcement Learning (RL). Among machine learning methods, RL has shown to be suitable for WSNs since it is fully distributed and its implementation requires minimal memory space and communication overhead.

The sensor network is a multiagent learning system in which nodes can be considered as autonomous agents making distributed decisions, called *strategies*, on how to use the shared wireless medium. An agent learns from interactions with its environment, including the other agents, which strategies to play in order to improve its own long-term reward. When needed, a node has to compete with its neighbours for access to the shared radio medium. Moreover, the intended receiver node should be ready to listen on the same channel and thus not to be transmitting its own data. This translates to learning which action to perform, that is, *transmit*, *listen*, or refrain from communication (or *sleep*)—in each time slot. Through the *success* or *failure* of their actions, the nodes learn to adapt their strategies to coordinate the transmissions on the appropriate channels.

A formal description of our multiagent learning system consists of the following components.

- (i) Each sensor node, denoted as n , is considered an autonomous agent that has a set of parent nodes, denoted as P_n , which can relay data of node n towards the sink.
- (ii) The number of time slots in a frame is denoted as K and the current slot index is denoted as k , with $0 < k \leq K$.
- (iii) A pseudorandom number generator function $c(n, k)$ is available to determine on which channel node n will listen in slot k .
- (iv) In each slot k , node n can derive the set of default channels; its parent nodes will listen on $C_{nk} = \{c(p, k) \mid p \in P_n\}$.
- (v) In each slot k , node n executes an action a_{nk} from its set of available actions A_{nk} : *listen* on its own channel $c(n, k)$ (denoted as action a_0 in Figure 3), or *transmit* on one of its parents' default channels c_x in C_{nk} (denoted as action a_x in Figure 3).
- (vi) For each slot k , node n keeps track of the probabilities of successfully performing respective actions in that slot, $P_{\text{suc}}(a_x)$ for all $a_x \in A_{nk}$.

Note that the sink, equipped with a half-duplex transceiver as well, listens in each slot for incoming data.

The flowchart in Figure 3 depicts how the learning process operates in every slot. Initially, each node chooses an action at random for all slots. In each slot, the nodes perform their actions simultaneously and the combined actions of all nodes result in a specific state of the network. Some pairs of nodes might have successfully established communication, while the actions of other nodes are unsuccessful due to collisions or unresolved coordination. Note that the data

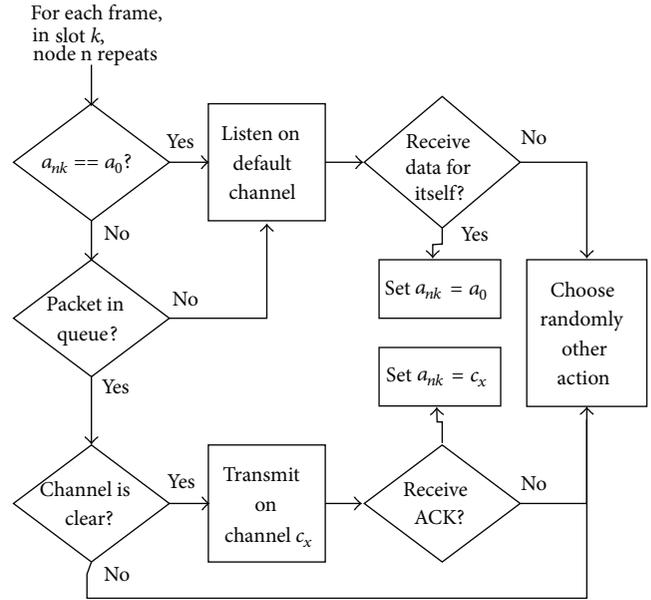


FIGURE 3: The learning process for scheduling collision-free transmission, which is operated in all nodes. The process is repeated every frame.

TABLE 1: Overview of why actions might fail.

Action	Reason for failure
Listen	Nobody sending (idle listening), collision
Transmit	Receiver not listening (deafness problem), collision (including hidden terminal)

mentioned in Figure 3 is the pseudo-data packet and the acknowledgement packet is the confirmation of receiving the pseudo-data packet.

When a node's action is successful, it will receive positive feedback either in the form of a data packet destined for itself (when listening) or an acknowledgment (when transmitting). The reason of a failed action is listed in Table 1.

To update the actions of the nodes, a trade-off between exploitation and exploration is made based on the "win-stay lose-shift" policy proposed in [19]. A successful action will be repeated in the same slot in the next frame ("win-stay"), while an action from the set of available actions will be randomly selected when the executed action failed ("lose-shift"). Different methods of action selection, based on a *uniform* or *biased* probability distribution, can be applied. In the uniform scheme, all actions in the set of available actions have equal selection probability. With the biased scheme, the probability of an action being chosen as the next action to perform is exponentially proportional to the probability of successfully performing that action (a parameter which has been updated in every frame). The policy used to balance exploitation and exploration influences to some extent on the optimality of the found solution.

To assure that nodes do not interfere with an already successfully coordinated communication, nodes perform a Clear Channel Assessment (CCA) check before transmitting.

At the beginning of each slot, a contention window is used in which the CCA is performed. The length of this contention window is decreased when a node has previously established successful transmissions in that slot. The mechanism will thus give privilege to nodes that performed successful transmissions before.

Individual nodes are not able to detect when the learning mechanism has converged, since it is fully distributed and nodes do not exchange their state information. The learning process is stopped after a fixed number of frames, set by the user of the system. After the learning process, the best action in each slot is chosen based on the success probabilities of the actions, in the following way:

$$\begin{aligned}
 & \text{select } a^* \text{ with } P_{\text{suc}}(a^*) > P_{\text{suc}}(a_x) \quad \forall a_x \in A_{nk}. \\
 & \text{If } P_{\text{suc}}(a^*) > P_{\text{th}} \text{ then} \\
 & \quad a_{nk} \leftarrow a^* \\
 & \text{else} \\
 & \quad a_{nk} \leftarrow \text{sleep} \\
 & \text{end if.}
 \end{aligned} \tag{1}$$

If the success probabilities of all actions are lower than or equal to a given threshold value (P_{th}), then the node infers that it cannot communicate successfully in that slot. The node can then turn off its radio in that slot, that is, *sleep*, to save its energy resource. The threshold value P_{th} is fixed by the user of the WSN.

5. Protocol Design

Time is discretized into fixed length frames composed of a specified number of slots. To achieve energy-efficient operation, the nodes duty cycle their radios in each slot. The length of the active period at the beginning of the slot is fixed and is long enough to allow a single IEEE 802.15.4 maximum length packet (of 128 bytes) to be transmitted and acknowledged. In every frame, each node periodically switches its channel at each time slot according to its pseudo-randomized channel-hopping sequence.

The protocol is divided into 3 phases as follows.

Initialization Phase. All nodes initially operate on the same default channel to allow initial synchronization and neighbour discovery. During network initialization, a minimum hop cost field is built. Each node learns its minimum hop distance to the sink and knows its set of candidate parent nodes through which it can relay data. If network density is high enough, this set will include more than one parent.

Scheduling Phase. Each node independently runs the learning process described in Section 4. Nodes learn how to coordinate their transmissions to achieve the goal of transferring the generated data message toward the sink while saving energy from failed transmissions and receptions. The achieved result

is the schedule of actions (listen, transmit on a specified channel, or sleep) for all nodes in each time slot.

Operation Phase. According to the obtained schedule after the learning phase, nodes know whether they should turn the radio off or on and which channel to switch to. They all follow the strategies which the learning process converged to. The main advantage of the proposed scheme, the characteristic of Reinforcement Learning technique, is that nodes do not need to share their own schedule to others. The coordination among nodes is achieved without central authorization or completely sharing information.

The obtained nodes' schedule is considered an "optimal" solution when all the generated packets in a frame can be delivered successfully at the sink within a frame. The schedule of each node is traffic adaptive since nodes only contend for channel access when they have packets in their queue (see Figure 3). The traffic-adaptive mechanism can be illustrated by the behaviour of the leaf nodes. Since the leaf nodes do not have to forward any data, they only need to transmit in a specific number of slots according to their own traffic load. In all other slots, the leaf nodes will enter in *sleep* mode and turn their radios off in order to avoid idle listening and overhearing.

In the proposed protocol, nodes do not need to exchange RTS/CTS messages. Since a node either wins access to the channel (and hence does not need RTS/CTS) or fails and then attempts to find another free slot during the scheduling phase, nodes actively avoid the deafness and hidden-terminal problems.

6. Experimental Study

6.1. Evaluated Parameters. A Matlab simulation program was developed to evaluate the performance of the proposed multichannel protocol. The following metrics are measured.

- (i) The end-to-end packet delivery ratio (PDR) is the ratio of the number of packets delivered at the sink to the total number of generated packets. This parameter indicates the optimality of the schedule obtained by the scheduling algorithm.
- (ii) The end-to-end latency is the average of the end-to-end delay of all packets received by the sink. Combined with the end-to-end PDR, this indicates the achieved network throughput.
- (iii) The energy waste factor is measured as the average amount of collisions, overhearing, and idle listening happening in one node per frame during the operation phase.

We compare the performance of our protocol with the multiple rendez-vous McMAC which is a frequency-hopping protocol.

We define the parameter α as the ratio of the number of time slots per frame to the number of generated data messages at all source nodes in the network. We investigate the relationship between the parameter α and the probability of converging to an optimal solution.

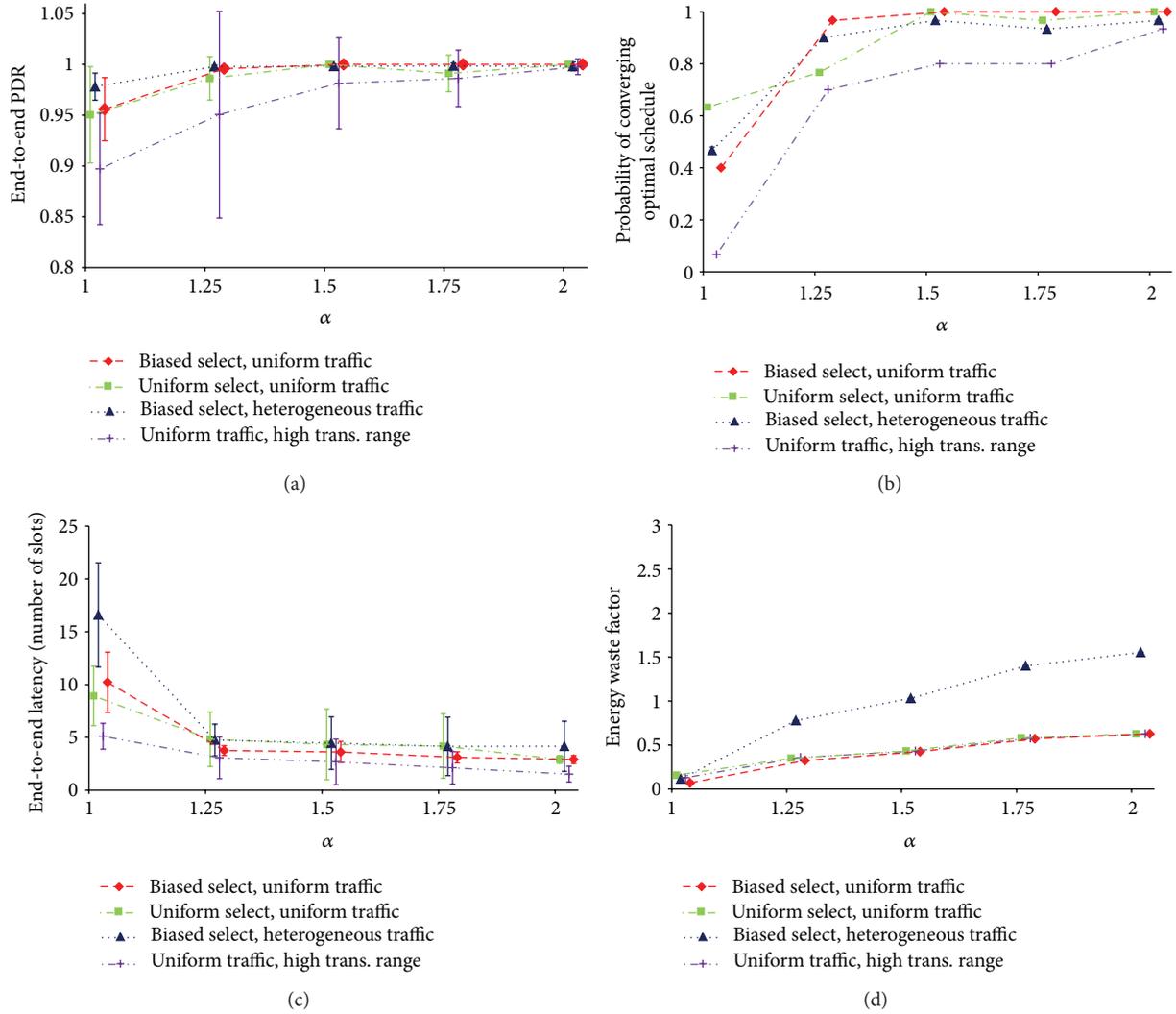


FIGURE 4: The performance evaluation in terms of (a) end-to-end packet delivery rate, (b) the probability of converging to optimal solution, (c) end-to-end latency, and (d) energy waste factor. The experiments have been done on grid topology of 25 nodes, in different settings, uniform or biased selection rule, uniform or heterogeneous traffic pattern, and different levels of transmission range.

We run simulation experiments with 25 nodes which are placed in an area of 80 m * 80 m in two kinds of topologies, grid and random topology. The sink is positioned at the leftmost upper corner. Nodes are randomly deployed in the area or uniformly deployed at a distance of 20 meters from each other. We set the transmission range of nodes at two levels, 30 meters and 50 meters.

To illustrate the performance of the network at high data rates, we assume that each node generates messages every 10 seconds. The length of the frame is set to 10 seconds. Each node generates 1 message per frame (uniform pattern) or 1 to 4 messages per frame (heterogeneous pattern). Messages are generated at random time slots within frame.

The scheduling phase is set to 200 frames, which is long enough for the algorithm to converge. A good value for the success probability threshold P_{th} was empirically determined to be 0.4. The results are averaged over 30 simulation runs and are represented as function of the parameter α .

6.2. Evaluation. As illustrated in Figure 4(a), when α is increased, the average of end-to-end PDR is increased. In grid topology, uniform or heterogeneous traffic pattern, and biased selection rule, the average of end-to-end PDR is larger than 99.5% when α is larger than 1.25. Even though, the error bar of standard deviation is very small (less than 0.5%). The result indicates that in most simulation runs, a contention-free schedule capable of delivering all messages to the sink has been achieved by the RL based scheduling algorithm. The probability of converging to such optimal solution is nearly 100%, as depicted in Figure 4(b). When the number of time slots per frame is equal to the number of generated messages (α equals to 1), the nodes have less possibilities to arrange their transmission.

The high value of end-to-end PDR and high probability of converging to an optimal solution in heterogeneous traffic pattern show that the attained schedule is adaptive with the heterogeneous traffic demand of each node. It comes from

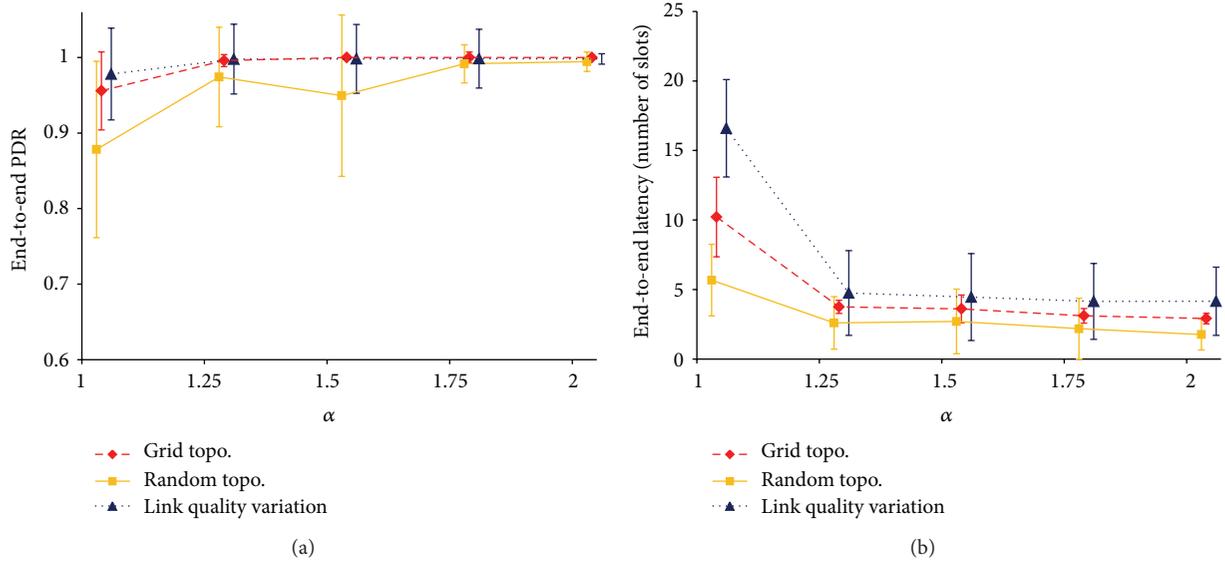


FIGURE 5: Performance evaluation of 25 nodes on random and grid topology and wireless link quality variation environment (average packet loss of 10%): (a) end-to-end packet delivery rate and (b) end-to-end latency.

the fact that a node only contends for the channel when it needs to send a message; otherwise, it listens or refrains from using the channel to save its own power and to give opportunity to others.

An increased communication range imposes a greater interference region as well. The nodes will thus have to overcome more constraints when they try to learn a collision-free schedule. Increasing the number of time slots per frame will alleviate this problem (when the number of parent nodes is about 3 to 5 in a grid topology with communication range of 50 meters, a good value for α is about 1.7).

The results of applying different rules for exploration, uniform or biased selection probability, show that the optimality of the obtained schedule is influenced by the exploration policy. With the uniform selection rule, the result is worse, lower average value and larger standard deviation of the end-to-end PDR parameter, which means that the attained solutions are often not optimal (not all nodes find the complete paths to send their messages). Figure 4(b) shows that the probability of converging to an optimal solution is a bit lower. The reason is that with a uniform selection rule, each action has equal probability to be chosen as an explored action. Nodes do not exploit the accumulated knowledge obtained during the learning process, which can be the number of successfully done trials of each action in the proposed scheme, which is used in biased probability selection rule. Therefore, the results are more fluctuating and not optimal.

Figure 4(c) shows the average number of time slots that a packet needs to be delivered to the sink. Equivalently to Figure 4(a), the performance is better when α is larger. For example, in grid topology, when the transmission range is set at 30 meters, the maximum hop count is 4 hops. The packets from the fourth tier require 4 slots to be delivered to the sink while those from the first tier require only 1 slot, which is the optimal achievable latency.

The energy waste factor of each test scenario in Figure 4(d) illustrates the energy expenditure in useless activities of nodes. The higher the value of α is, the higher the energy waste becomes. In detail, nodes mainly waste their energy on the idle listening. The overhearing rarely happens and collisions are only found in nonoptimal schedules.

Figure 5 shows the performance comparison in different topologies, grid and random topology, and in link quality variation environment. In a random topology, a somewhat lower performance is perceived even when the number of slots per frame is high ($\alpha = 2$). The reason is that in random topology, the numbers of neighbour nodes and parent nodes are very different from node to node. Therefore, a fixed sleep threshold (P_{th}) appears unsuitable. We presume that a variable sleep threshold in relation with the number of parent nodes of each node will be more appropriate.

In real-world application, the quality of wireless link often varies, which means that the average of packet delivery rate at each link is less than 100%. We set up a simulation in which the average packet loss ratio of every link is 10%. To compensate for packet loss, the retransmission scheme is applied to assure reliable communication. In the learning phase, we set up each node to learn with the double of its own generated traffic. Hence, nodes contend more for access channel and might get more slots for transmissions. It would help nodes to have redundancy slots for retransmission if the transmission has failed because of low link quality. The effectiveness of the redundancy scheme for link variation has been proved by the very high end-to-end PDR and the one frame-bounded end-to-end delay illustrated in Figure 5.

The proposed protocol outperforms McMAC since McMAC does not provide contention-free access (see Figure 6). The receiver-centric contention resolution of McMAC is not enough to assure a decent packet delivery ratio. The average end-to-end latency is much higher because

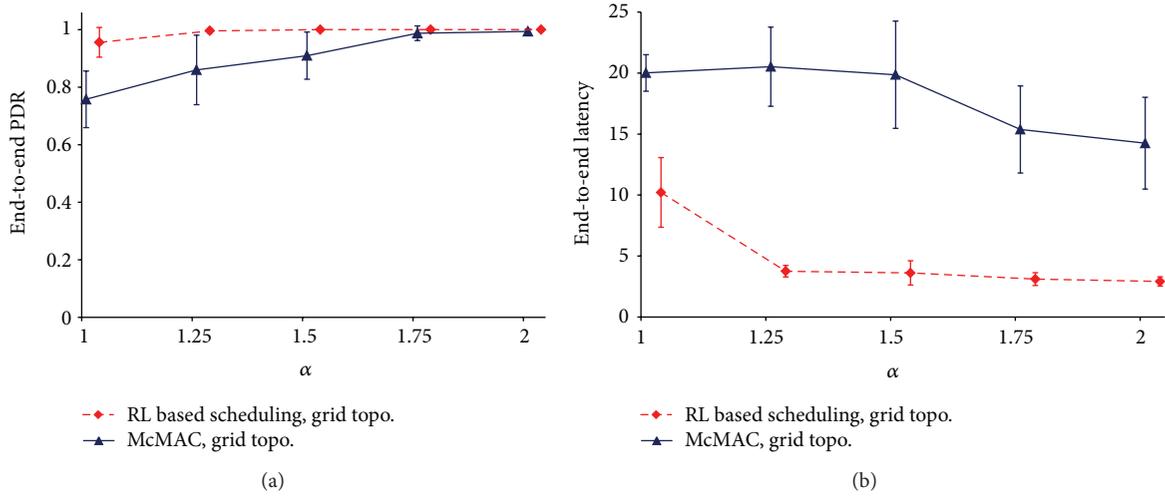


FIGURE 6: Comparison of Reinforcement Learning based scheduling in multichannel protocol with McMAC on grid topology of 25 nodes, traffic pattern of one packet per frame in terms of (a) End-to-end packet delivery rate and (b) End-to-end latency.

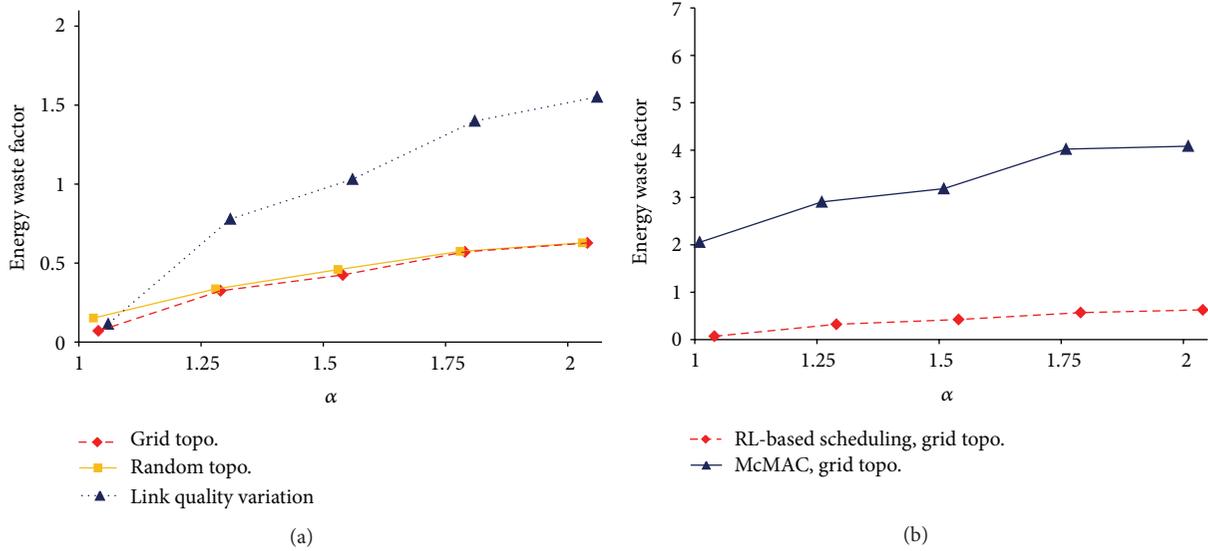


FIGURE 7: The energy waste factor of (a) Reinforcement Learning based scheduling in multichannel protocol on random or grid topology and link quality variation scenarios and (b) in comparison with McMAC.

of CSMA/CA scheme in McMAC. The proposed collision-free schedule protocol especially shows a very impressive energy efficiency compared to McMAC (see Figure 7(b)). For example, in McMAC protocol, on average a node has wasted its energy in 3 time slots every frame of 36 time slots ($\alpha = 1.5$). In the proposed protocol, the result is 1 time slot every 3 frames, which is 9 times less than McMAC. In the adaptive learning scheduling protocol, most of energy waste comes from idle listening, while in McMAC, the collision, idle listening, and overhearing all happen much more frequently.

7. Discussion

The experimental results indicate that the performance of the proposed protocol depends on the parameter α , which is

the ratio of the number of slots per frame to the number of generated data messages per frame. When α is large enough, the optimal solution is achieved more easily and more often, latency is lower and less packet loss is experienced. However, increasing α will result in more idle listening. When the number of slots per frame is larger, nodes might try to transmit in more slots before converging to a specified one to make some receiver node conclude that the action of *listening* is a good one. In addition, the conclusion of nodes also depends on the sleep threshold. To ensure a high packet delivery rate, the parameter of sleep threshold (P_{th}) should not be set too large; otherwise, a lot of nodes might decide to sleep. The value of P_{th} reported in Section 6 is obtained experimentally. Further investigation should be done to explore this issue.

The proposed protocol is based on schedule-based approach, which will require network-wide time synchronization which has been a well-investigated research topic in wireless sensor networks since it is essential for any kind of distributed components [20]. Among that, Time-Synch Protocol for Sensor Networks (TPSNs) [21] providing network-wide time synchronization with a master clock is emerging as a promising protocol.

In TPSN, the hierarchical-structured tree is used to distribute the global time of the master clock at the sink. Along the routing tree, nodes, receiving timing information, estimate their clock offset with the master clock and adjust their clock information when needed. Hence, nodes are time synchronized with the sink node. In [22], the experimental result on real testbed of a derivation of TPSN has proved the feasibility of tight requirement in synchronization for schedule-based approach protocol with reasonable overhead.

We observe that the operation of TPSN is suitable for our schedule-based multichannel protocol. The network is also built in a tree form in which each node knows its minimum distance to the sink. The timing information from the sink can be distributed to source nodes in broadcast slots. During periodic traffic forwarding to the sink, a sender node can receive timing information from the receiver which is its parent node in the routing tree and keep synchronized. Future work will present the detailed extension of TPSN for a practical time synchronization scheme supporting the proposed protocol operation on real hardware.

8. Conclusions

In this paper, we proposed a multichannel protocol for high bandwidth wireless sensor networks. The proposed protocol is based on the combination of TDMA and FDMA and provides collision-free scheduling of transmissions by applying a Reinforcement Learning technique for the joint scheduling and routing process in each node. The joint problem comes from the motivation that a fixed routing structure may result in suboptimal exploitation of the multiple channels. The medium access resolution is combined with route selection to make nodes learn not only to which parent but also on which channel they should forward their data. The scheduling algorithm works in a distributed manner on each node, resulting in an optimal solution of collision-free transmission for all nodes. Moreover, the obtained schedule is traffic adaptive and energy efficient.

We have investigated the performance of the proposed protocol with extensive simulation experiments. The results show that the optimal solution is obtained in nearly 100% of the simulation runs when the number of time slots per frame is greater than the traffic demand.

Taking into account wireless link quality variation, redundant traffic pattern learning is applied. It assures high end-to-end packet delivery rate and bounded end-to-end latency.

In comparison with McMAC, another frequency-hopping approach, our proposed protocol shows a better performance in terms of end-to-end delivery rate, end-to-end latency, and much higher energy efficiency.

In the future, we plan to implement the proposed protocol in combination with the appropriate time synchronization protocol on an operational sensor node testbed and evaluate its performance on the real devices.

Acknowledgments

This work was performed in the scope of FWO project G.0291.09N: QoS support in heterogeneous wireless sensor networks using distributed Reinforcement Learning techniques. K.-Ha. Phung is grateful to the Belgian Technical Cooperation (BTC) for granting her a mixed Ph.D. Scholarship.

References

- [1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "Wireless multimedia sensor networks: applications and testbeds," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1588–1605, 2008.
- [2] A. Raniwala and T. C. Chiueh, "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, pp. 2223–2234, Miami, Fla, USA, March 2005.
- [3] J. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: handling multi-channel hidden terminals using a single transceiver," in *Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MoBiHoc '04)*, pp. 222–233, Tokyo, Japan, May 2004.
- [4] B. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar, "Component based channel assignment in single radio, multi-channel ad hoc networks," in *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM '06)*, pp. 378–389, Los Angeles, Calif, USA, September 2006.
- [5] P. Bahl, R. Chandra, and J. Dunagan, "SSCH: slotted seeded channel hopping for capacity improvement in IEEE 802.11 ad-hoc wireless networks," in *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking (MobiCom '04)*, pp. 216–230, New York, NY, USA, October 2004.
- [6] H. Sheung, W. So, J. Walrand, and J. M. McMAC, "A parallel rendez-vous multi-channel mac protocol," in *Proceedings of Wireless Communications and Networking Conference (WCNC '07)*, pp. 334–339, October 2007.
- [7] G. Yu and C. Chang, "An efficient cluster-based multi-channel management protocol for wireless ad hoc networks," *Computer Communications*, vol. 30, no. 8, pp. 1742–1753, 2007.
- [8] Y. Wu, J. Stankovic, T. He, and S. Lin, "Realistic and efficient multi-channel communications in wireless sensor networks," in *Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM '08)*, pp. 1193–1201, 2008.
- [9] Q. Yu, J. Chen, Y. Fan, X. Shen, and Y. Sun, "Multi-channel assignment in wireless sensor networks: a game theoretic approach," in *Proceedings of the 29th IEEE Conference on Computer Communications (INFOCOM '10)*, pp. 1–9, March 2010.
- [10] M. Salajegheh, H. Soroush, and A. Kalis, "HYMAC: Hybrid TDMA/FDMA medium access control protocol for wireless sensor networks," in *Proceedings of the 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '07)*, pp. 1–5, September 2007.

- [11] O. D. Incel, L. van Hoesel, P. Jansen, and P. Havinga, "MC-LMAC: a multi-channel MAC protocol for wireless sensor networks," *Ad Hoc Networks*, vol. 9, no. 1, pp. 73–94, 2011.
- [12] A. Förster and A. L. Murphy, "Froms: a failure tolerant and mobility enabled multicast routing paradigm with reinforcement learning for WSNs," *Ad Hoc Networks*, vol. 9, no. 5, pp. 940–965, 2011.
- [13] M. Mihaylov, Y. A. le Borgne, K. Tuyls, and A. Nowé, "Decentralised reinforcement learning for energy-efficient scheduling in wireless sensor networks," *International Journal of Communication Networks and Distributed Systems*, vol. 9, pp. 207–224, 2012.
- [14] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, The MIT Press, Cambridge, Mass, USA, 1998.
- [15] E. J. Duarte-Melo and M. Liu, "Data-gathering wireless sensor networks: organization and capacity," *Computer Networks*, vol. 43, no. 4, pp. 519–537, 2003.
- [16] O. D. Incel, "A survey on multi-channel communication in wireless sensor networks," *Computer Networks*, vol. 55, no. 13, pp. 3081–3099, 2011.
- [17] <http://www.ti.com/product/cc2420>.
- [18] J. Li, Z. Haas, M. Sheng, and Y. Chen, "Performance evaluation of modified IEEE 802.11 MAC for multi-channel multi-hop ad hoc network," in *Proceedings of the 17th International Conference on Advanced Information Networking and Applications (AINA '03)*, pp. 312–317, 2003.
- [19] M. Posch, "Win-stay, lose-shift strategies for repeated games—memory length, aspiration levels and noise," *Journal of Theoretical Biology*, vol. 198, no. 2, pp. 183–195, 1999.
- [20] I. K. Rhee, J. Lee, J. Kim, E. Serpedin, and Y. C. Wu, "Clock synchronization in wireless sensor networks: an overview," *Sensors*, vol. 9, no. 1, pp. 56–85, 2009.
- [21] S. Ganeriwal, R. Kumar, and M. B. Srivastava, "Timing-sync protocol for sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems*, pp. 138–149, Los Angeles, Calif, USA, November 2003.
- [22] B. Lemmens, K. Steenhaut, P. Ruckebusch, I. Moerman, and A. Nowe, "Network-wide synchronization in wireless sensor network," in *Proceedings of 19th IEEE Symposium on Communication and Vehicle Technology in the Benelux (SCVT '12)*, Eindhoven, The Netherlands, November 2012.

