Duty-Cycle Optimization for IEEE 802.15.4 Wireless Sensor Networks

Pangun Park, University of California at Berkeley
Sinem Coleri Ergen, Koc University
Carlo Fischione, KTH, Royal Institute of Technology
Alberto Sangiovanni-Vincentelli, University of California at Berkeley

Most applications of wireless sensor networks require reliable and timely data communication with maximum possible network lifetime under low traffic regime. These requirements are very critical especially for the stability of wireless sensor and actuator networks. Designing a protocol that satisfy these requirements in a network consisting of sensor nodes with traffic pattern and location varying over time and space is a challenging task. We propose an adaptive optimal duty-cycle algorithm running on top of the IEEE 802.15.4 medium access control to minimize the power consumption while meeting the reliability and delay requirements. Such a problem is complicated because simple and accurate models of the effects of the duty-cycle on the reliability, delay, and power consumption are not available. Moreover, the scarce computational resources of the devices and the lack of prior information about the topology make it impossible to compute the optimal parameters of the protocols. Based on an experimental implementation, we propose simple experimental based models to expose the dependency of reliability, delay, and power consumption on the duty-cycle at the node and validate it through extensive experiments. The coefficients of the experimental based models can be easily computed on existing IEEE 802.15.4 hardware platforms by introducing a learning phase without any explicit information about data traffic, network topology and medium access control parameters. The experimental based model is then used to derive a distributed adaptive algorithm for minimizing the power consumption while meeting the reliability and delay requirements in the packet transmission. The algorithm is easily implementable on top of the IEEE 802.15.4 medium access control without any modifications of the protocol. An experimental implementation of the distributed adaptive algorithm on a test-bed with off-the-shelf wireless sensor devices is presented. The experimental performance of the algorithms is compared to existing solutions from the literature. The experimental results show that the experimental based model is accurate and the proposed adaptive algorithm attains the optimal value of the duty-cycle maximizing the lifetime of the network while meeting the reliability and delay constraints under both stationary and transient conditions. Specifically, even if the number of devices and their traffic configuration change sharply, the proposed adaptive algorithm allows the network to operate close to its optimal value.

Categories and Subject Descriptors: C.2.2 [Computer-Communication Networks]: Network Protocols
General Terms: Performance, Standardization, Experimentation, Theory

ACM Reference Format:
DOI = 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

Pangun Park and Alberto Sangiovanni-Vincentelli are with the University of California at Berkeley, EECS Department, Berkeley, CA, e-mail: {pgpark|alberto}@eecs.berkeley.edu. Sinem Coleri Ergen is with Electrical and Electronics Engineering, Koc University, Istanbul, Turkey, e-mail: sergen@ku.edu.tr. Carlo Fischione is with the Automatic Control Lab, ACCESS Linnaeus Center and Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden, e-mail: carlofi@kth.se. Sinem Coleri Ergen acknowledges the support the Marie Curie Reintegration Grant IVWSN. Carlo Fischione acknowledges the support the Swedish Research Council, the EU STREP project HydroBioNets, and the EU NoE Hycon2.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee.
Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© YYYY ACM 1539-9087/YYYY/01-ARTA $15.00
DOI 10.1145/0000000.0000000 http://doi.acm.org/10.1145/0000000.0000000

ACM Transactions on Sensor Networks, Vol. V, No. N, Article A, Publication date: January YYYY.
1. INTRODUCTION

Many applications using wireless sensor networks (WSNs) require a certain degree of the probability of successful packet reception (reliability) and timely data communication to the collection center under low traffic regime. These requirements are critical particularly for the stability of the WSN based control and automation applications [Willig et al. 2005]. In these applications, if reliability and delay requirements are not met, the correct execution of control actions or decisions concerning the phenomena sensed may be severely compromised. Satisfying high reliability and low delay requirements however may demand significant power consumption. Maximizing reliability or minimizing delay is usually not an optimal design strategy: reliability and delay must be flexible design parameters that need to be adequate for the application requirements while minimizing the power consumption to ensure long lifetime of the network [Zhang et al. 2001]. Energy efficiency is critical for applications with battery-powered devices. The radio in WSNs consumes a considerable amount of energy and listening to the radio channel consumes as much energy as receiving data. Idle listening should be minimized since it does not contribute to the operation of the network, yet it may require a large amount of energy.

Several duty-cycle protocols have been proposed as an effective mechanism for reducing idle listening (see, e.g., GAF [Xu et al. 2001], SPAN [Chen et al. 2001], SMAC [Ye et al. 2004], and low-power-listening (LPL) [Hill and Culler 2002]). Such protocols are based on periodical cycling between a sleep and a listening state. Key parameter determining the duty-cycle is the sleep time for a given listening time. The main advantage of duty-cycling is that nodes do not require any additional hardware such as a wake-up radio [Guo et al. 2001]. Even more importantly, it does not require complex control mechanisms, as in time division multiple access (TDMA) schemes, for discovering network topology, keeping the nodes synchronized [Coleri-Ergen and Varaiya 2006] and running the schedules efficiently [Uysal-Biyikoglu et al. 2002]. Duty-cycling is particularly appealing for dynamic networks where the locations of the sensor nodes and data traffic generated at each node are changing over time [Jurdak et al. 2010]. However, the intrinsic simplicity of the mechanism has the drawback of smaller energy saving potential as compared to the more complex solutions listed above unless the duty-cycling is adapted to changes of data traffic and network topology.

Duty-cycling medium access control (MAC) protocols are of two types: synchronous and asynchronous. Synchronized protocols such as SMAC [Ye et al. 2004], TMAC [Van Dam and Langendoen 2003], WiseMAC [El-Hoiydi and Decotignie 2004] and SyncWUF [Shi and Stromberg 2007] are based on negotiating a schedule among the neighbors to specify when the nodes are sleep and awake. Asynchronous protocols on the other hand are based on preamble sampling which was first introduced as the well known LPL in [Hill and Culler 2002] and then followed by many protocols that have a similar concept including BMAC [Polastre et al. 2004] and X-MAC [Buettner et al. 2006]. In this method, the receiver wakes up periodically to check whether there is a transmission and the sender, instead of coordinating the neighbors’ wake up times, sends a preamble that is long enough to ensure the receiver wakes up during the preamble. Asynchronous protocols are more popular in practice since the simple mechanism does not require either global synchronization or topology knowledge [Bachir et al. 2010; Langendoen and Meier 2010]. In this paper, we focus on the asynchronous preamble sampling protocol.

Despite its successful usages, asynchronous protocols have one fundamental question to answer as to whether the duty-cycle gives good performance for applications. Lowering the duty-cycle implies putting nodes in sleep mode for larger periods. While using a larger sleep time reduces the cost of idle listening at the receiver, it increases the transmission cost as the transmitter uses a longer preamble. Hence, there is a tradeoff between the receiving cost of idle listening and transmission cost of longer preamble. Furthermore, as the sleep time increases, the reliability, throughput, and delay significantly degrade due to the high contention in the medium with increasing traffic. The tradeoff between power consumption, reliability, and delay of the network should be adjusted based on the application requirements.
We explicitly consider the random access mechanism of the unslotted IEEE 802.15.4 protocol to improve the reliability and delay performance of preamble sampling protocols. The IEEE 802.15.4 standard has received considerable attention as a low data rate and low power consumption protocol for WSN applications in industry, control, home automation, and health care [IEEE 2006]. It has been adopted with minor variations also by other protocols such as ZigBee [Wheeler 2007] and ISA100 [ISA 2009]. We remark that the unslotted IEEE 802.15.4 protocol is not energy efficient since there is no explicit mechanism to save energy consumption. It is natural to combine the duty-cycle mechanism and the unslotted IEEE 802.15.4 protocol. However, it is not trivial to find the optimal duty-cycle because this optimal value depends upon several parameters such as random access mechanism, traffic load, network topology, and hardware specifications, and needs to consider the reliability and delay requirements of applications while minimizing power consumption.

The goal of this paper is to design an adaptive duty-cycle algorithm to achieve maximum lifetime while guaranteeing the reliability and delay constraints of the application. We focus on how to tune the duty-cycle for IEEE 802.15.4 MAC instead of designing an entirely new asynchronous duty-cycle protocol. Solving a duty-cycling optimization problem requires a number of cost and constraint function evaluations. Unfortunately, the dependence of these functions on the design parameters is implicit and quite complicated [Fischione et al. 2009]. Consequently, solving the optimization problem online is out of the question if we use the full fledged model given the limited computational resources of the nodes. The most important problem here is finding the tractable model of the optimization without significant loss of accuracy. Our work is inspired by the simple observation of the dependency of reliability and delay for different traffic loads of the network: Without loss of generality, as the traffic load decreases, the linear factor of the reliability and delay dependence on the sleep time become dominant than the nonlinear factor. The original contributions of this paper are three:

(1) We demonstrate the existence of a linear relation between reliability, delay of the packets and sleep time, and a quadratic relation between power consumption and sleep time for a given listening time under the low traffic regime. The effect of listening time on reliability and delay is negligible as we increase listening time above a certain value. A simple method can estimate the coefficients of these experimental based models without requiring a high computational load.

(2) We propose an adaptive optimal duty-cycle (AODC) algorithm for unslotted IEEE 80215.4 standard to minimize the power consumption while meeting the reliability and delay requirements. The proposed algorithm explicitly considers the random access mechanism of the standard.

(3) The proposed AODC algorithm is implemented on a test-bed using TelosB sensors [Polastre et al. 2005]. Experimental results show that this algorithm meets reliability and delay requirements while achieving the high power efficiency under both stationary and transient condition of the network.

The rest of the paper is organized as follows: Section 2 gives an overview of existing studies. Section 3 presents the system model. In Section 4 the optimization problem is formulated and the challenges in solving this problem are stated. Section 5 validates a simple experimental based model through extensive experiments. In Section 6, the solution of the optimization problem is presented and the adaptive algorithm to implement the solution is described. Numerical results achieved during stationary and transient conditions are reported in Section 7. Finally, Section 8 concludes the paper.

2. RELATED WORK

B-MAC [Polastre et al. 2004] is an asynchronous preamble sampling protocol extending LPL technique by a user-controlled sleep interval. Each node independently repeats a sleep/active cycle without negotiating on the schedules. When transmitter sends a data packet, it sends a preamble long enough to cover one complete sleep interval, which assures that the receiver can detect the signal and eventually the start symbol, followed by the data message. The X-MAC [Buettner et al. 2006] is a refinement of B-MAC for packet-based radios. The transmitter sends a packet strobe instead
of sending a long wake-up preamble of B-MAC. Once the node receives the right packet strobe, it replies with an ACK. Then, the data message exchange takes place immediately. This ACK mechanism in X-MAC reduces the average preamble transmission time so the time, energy and overhead to transfer a data packet since the entire preamble does not need to be sent if the receiver was already awake. Therefore, we integrate X-MAC with the unslotted IEEE 802.15.4 protocol in this paper. X-MAC also includes a lookup table to adapt the duty-cycle of the nodes based on the traffic load. However, the proposed solution is suboptimal since the random access mechanism is not considered in the optimization problem. Moreover, no delay or reliability constraint on packet delivery is considered, which means that the energy minimization proposed by X-MAC does not guarantee any timely successful packet delivery.

The idea of adaptive duty-cycling of preamble sampling protocol is presented in [Jurdak et al. 2010], where the authors use the energy consumption of each node in the routing decision of the cross-layer solution. Each node determines the preferred parent in the routing tree based on the routing cost that is a function of the ratio of the duty-cycles of neighbors to the average duty-cycle in the neighborhood. The duty-cycle is then chosen proportional to the expected number of packets to transmit. This model however does not consider the reliability and delay requirements nor minimizes the power consumption.

In [Park et al. 2009], the authors derive the energy consumption of a node as a function of the duty-cycle. This model is then applied to formulate two optimization problems, one minimizing the total energy consumption and the other maximizing the network lifetime. These problems are solved by using an iterative algorithm that requires the global topology information. The analytical model of the energy consumption however does not take the collision and contention in sending a packet, the random access mechanism, the packet copy delay, and the delay to tune the transceiver into account. The proposed practical heuristic algorithms are based on the exchange of the information of the energy consumption of neighbors. The algorithms tune the duty-cycle by following additive increase/additive decrease (AIAD) policy based on either the variation in the total energy consumption of the node itself and its neighbors or the comparison of its energy consumption with the maximum energy consumption of its neighbors. This study however does not consider either the delay nor reliability. Furthermore, the proposed algorithms are analyzed through the simulation without any experimental validations.

A dynamic sleep time control approach to reduce control packet energy waste that uses available statistical network traffic information has been proposed in [Ning and Cassandras 2010]. The authors propose two distinct approaches to dynamically compute the sleep time, depending on the objectives and constraints of the network. The first approach provides a dynamic sleep time policy that meets a specified average delay based on the packet waiting time. The second approach determines the optimal policy that minimizes total energy consumed. Both approaches require the interarrival time distribution of traffic loads. However, in practice, the network traffic information is not usually known in advance. Therefore, this paper presents a quantile-based distribution approximation and learning algorithm to estimate a probability distribution. This approach is computationally demanding because each node needs to estimate the interarrival time distribution of traffic loads and solve an optimization problem using numerical methods. In addition, the control packet overhead increases since each node sends the interarrival time of traffic loads. Furthermore, this sleep time control algorithm does not clearly describe the mechanism when it deals with many-to-one communication. In particular, the reliability issue of this algorithm is critical for many-to-one communication.

In [Merlin and Heinzelman 2010], two adaptive duty-cycle algorithms are presented to meet the target successful packet transmission rate while ensuring a longer lifetime of the network. The first algorithm, called asymmetric additive duty-cycle control (AADCC) [Merlin and Heinzelman 2010] is based on a linear increase/linear decrease of the duty-cycle depending on the comparison of the successfully received packet rate and its target value. Whenever five consecutive packets are successfully sent to the destination, the sleep time is increased by 0.1 s. Otherwise, each node decreases the sleep time by 0.25 s. The second algorithm, called dynamic duty-cycle control (DDCC), on the other hand aims to balance the reliability and energy consumption by using control theory. In DDCC,
Duty-Cycle Optimization for IEEE 802.15.4 Wireless Sensor Networks

Fig. 1. Clustered network topology. The packets generated by the gray nodes are transmitted to the sink node depicted in the middle of each cluster.

A simple control law is applied to adapt the sleep time for a deterministic noisy linear process representation of the network. Each node periodically updates the characteristics of the system model. The proposed algorithms are evaluated through Matlab simulation without any implementation due to the difficulty in measuring the energy consumption and computation load. Even though the number of estimator coefficients is reduced, the computation load makes it hard to run the algorithms in sensor nodes. In multi-hop networks, the algorithm requires time synchronization along the routing path, which can be very difficult and is in contrast with the simplicity of asynchronous duty-cycle protocol. Furthermore, the algorithm does not guarantee by design a minimum energy consumption, a desired delay and reliability in the packet delivery.

In [Cohen and Kapchits 2009; Kim et al. 2010], some analytical studies of the synchronous duty-cycle algorithms are presented by formulating different optimization problems. In [Cohen and Kapchits 2009], the authors pose the problem of determining the optimal duty-cycle for the minimization of the energy consumption for a maximum latency requirement. In [Kim et al. 2010], the authors propose a similar problem to minimize the delay and maximize the network lifetime of event-driven traffic pattern. However, the memory requirement and computation complexity to run these algorithms are still high for resource-constrained sensor nodes. An asynchronous random sleeping (ARS) mechanism is investigated in [Hua et al. 2007], whereby sensors wake up randomly and independently of others in each time slot to maximize the stationary coverage probability. The ARS offers statistical sensing coverage; its performance can be characterized by the stationary coverage probability and the coverage periods. The closed-form expressions of the stationary coverage probability, the expected k-coverage periods, and the expected k-vulnerable periods are derived using the renewal process theory. The homogenous wakeup probability is computed by using an analytical result. However, in general, the wakeup probability is heterogeneous, depending on the location, platform, and different application requirements. The papers [Cohen and Kapchits 2009; Kim et al. 2010; Hua et al. 2007] validate their algorithms via simulation without experiments.

The studies in [Hill and Culler 2002; Polastre et al. 2004; Buettner et al. 2006; Park et al. 2009] focus on the minimization of the energy consumption of the network. We remark that our target is to design an adaptive duty-cycle algorithm in order to minimize the power consumption while meeting the reliability and delay requirements. There is no adaptive duty-cycle protocol in the literature that considers all these aspects. In addition, these studies do not consider the random access mechanism of the unslotted IEEE 802.15.4 protocol and the packets are assumed to be always successfully received without collisions. Contrary to previous studies in [Hill and Culler 2002; Jurdak et al. 2010; Polastre et al. 2004; Buettner et al. 2006; Park et al. 2009; Ning and Cassandras 2010; Merlin and Heinzelman 2010; Cohen and Kapchits 2009; Kim et al. 2010], we consider the timely reliability rather than packet reception rate or the expected number of packets to transmit.

3. SYSTEM MODEL

We assume that the nodes of the WSN are organized into clusters as shown in Fig. 1. Clustered network is an essential topology for a number of standardization groups [IEEE 2010] and commercial
products [Wheeler 2007] such as asset tracking, process control, and building automation. Clustered network topology is supported in networks that require energy efficiency since it allows local data aggregation and eliminates the disadvantages of unbalanced energy consumption in multi-hop routing and high energy consumption of transmitting directly to the base station [Heinzelman et al. 2000]. In a clustered topology, nodes organize themselves into clusters with a node acting as the cluster head. All non-cluster head nodes transmit their data directly to the cluster head, while the cluster head receives data from all cluster members and transmits them to other cluster heads or a remote base station. Note that even such a simple topology presents highly challenging dynamics to model.

Throughout this paper we consider a probabilistic packet generation model rather than a periodic packet generation model, because the preamble sampling protocol is an asynchronous random access mechanism. We consider that the packet generation probability is uniformly distributed over the packet generation period. Given this source characteristic, the unslotted IEEE 802.15.4 is the natural MAC choice [IEEE 2006].

In preamble sampling protocols, the receiver wakes up periodically for a short time to sample the medium. When a sender has data, it transmits a series of short preamble packets, each containing the identifier of the target node, until it either receives an ACK packet from the receiver or a maximum sleep time is exceeded. Following the transmission of each preamble packet, the transmitter waits for the timeout. If the receiver is the target, it sends back an ACK. Upon reception of the ACK, the sender transmits the data packet to the destination. Fig. 2 shows the communication states between a transmitter and a receiver.

Coherently with the IEEE 802.15.4 standard in the unslotted modality, we assume that the data and preamble packets are sent using random access whereas the ACK frame is sent immediately upon reception of the preamble. For the packets that are sent using random access, the time duration between sending the packet to the MAC layer and over the physical link is random. In IEEE 802.15.4 standard, a node that sends a data frame shall wait for at most \( \text{macAckWaitDuration} \) for the corresponding ACK frame to be received. Hence, the timeout to receive an ACK is equal to \( \text{macAckWaitDuration} \) of the standard. Consequently, the maximum listening time is the sum of the timeout and maximum backoff time of the random access. The time duration in random access may be much larger than the packet transmission time. In IEEE 802.15.4 radios with default parameter settings, the maximum backoff before packet transmission is \( 2^{7} \times 4 = 27 \text{ ms} \) whereas the transmission time of a 56 byte packet is 1.79ms at 250kbps. The amount of random access, which depends on the data traffic, network topology and the parameters of the MAC protocol should therefore be included in the power minimization problem since random access

— determines the time interval between the transmissions of two consecutive preamble packets;
— determines listening time, since the destination node should receive at least one preamble packet during the listening time;
— is affected by sleep time, since increasing sleep time increases both the expected number of preambles in the network and the time duration spent in random access.
To illustrate the dependency of the random access on the data traffic, the network topology and the parameters of the MAC protocol, we briefly explain the random access mechanism in IEEE 802.15.4 protocol next.

3.1. IEEE 802.15.4 Unslotted CSMA/CA Mechanism

In the unslotted IEEE 802.15.4 carrier sense multiple access with collision avoidance (CSMA/CA) mechanism, each node in the network has two variables: \(NB\) and \(BE\). \(NB\) is the number of times the CSMA/CA algorithm has backed off while attempting the current transmission. \(NB\) is initialized to 0 before every new transmission. \(BE\) is the backoff exponent, which is related to how many backoff periods a node must wait before it attempts to assess the channel. The algorithm is implemented using units of time called backoff periods. The parameters that affect the random backoff are \(BE_{\text{min}}, BE_{\text{max}}\) and \(NB_{\text{max}}\), which correspond to the minimum and maximum of \(BE\) and the maximum of \(NB\) respectively.

The unslotted CSMA/CA mechanism works as follows. \(NB\) and \(BE\) are initialized to 0 and \(BE_{\text{min}}\) respectively (Step 1). The MAC layer delays for a random number of complete backoff periods in the range 0 to \(2^{BE} - 1\) (Step 2) and then requests PHY to perform a clear channel assessment (CCA) (Step 3). If the channel is assessed to be busy (Step 4), the MAC sub-layer increments both \(NB\) and \(BE\) by one, ensuring that \(BE\) is not more than \(BE_{\text{max}}\). If the value of \(NB\) is less than or equal to \(NB_{\text{max}}\), the CSMA/CA must return to Step 2. Otherwise, the CSMA/CA must terminate with a status of channel access failure. If the channel is assessed to be idle (Step 5), the MAC layer starts transmission immediately.

The expected number of random backoffs is a function of busy channel probability during channel sensing states, which depends on the channel traffic. Channel traffic on the other hand depends on data traffic, network topology and duty-cycling, since they determine the expected number of preamble packets. This complex interdependence is investigated in the following sections.

4. Protocol Optimization

The goal of our duty-cycle protocol is to find the optimal sleep time and listening time of each receiver node such that the overall power of the network is minimized under reliability and delay constraints. The formulation of the optimization problem is as follows:

\[
\begin{align*}
\min_{T_l, T_s} & \ E(T_l, T_s) \\
\text{s.t.} & \ R(T_l, T_s) \geq R_{\text{min}}, \\
& \ D(T_l, T_s) \leq D_{\text{max}},
\end{align*}
\]

where \(E(T_l, T_s)\) is the expected power consumption of the network, which includes transmit, receive, listen and sleep power, and \(T_l\) and \(T_s\) are the listening time and sleep time respectively. \(R(T_l, T_s)\) and \(D(T_l, T_s)\) are the expected reliability and delay of the network whereas \(R_{\text{min}}\) and \(D_{\text{max}}\) are the minimum acceptable reliability and maximum acceptable delay respectively. More specifically, the reliability is defined as the probability of successful packet reception, whereas the delay is defined as the time interval from the instant the packet is generated, until the transmission is successful after receiving the corresponding ACK from the receiver. The cost function and constraints are given by statistical expectation over time. The solution of the optimization problem gives the optimal sleep and listening times of the nodes. This optimization problem should be solved when the nodes are first deployed and in case of changes in the network topology or application requirements.

In general, the power consumption, reliability and delay depend on both \(T_l\) and \(T_s\). The exact computation of the analytical expressions in the optimization problem is a challenging task, since the duty-cycle of each node affects the reliability, delay and power consumption, along with the traffic load and network topology. Furthermore, the traffic load, channel condition, MAC parameters, and network topology affect the total backoff time of random access mechanism, which then determines the number of preambles together with the listening time and sleep time of the receiver.
Accurate analytical models of the expectations in problem (4) have been investigated in [Fischione et al. 2009]. Unfortunately, in these analytical models the relation among the decision variables is highly non linear, which would require the use of sophisticated optimization tools to solve problem (4). Clearly, this is difficult or impossible to implement in resource-constrained sensor nodes. To overcome these problems, we propose an experimental based model, where the cost function and the constraints of problem (4) are approximated based on the observations from an extensive set of experiments. We will see in the next section that the equations depend on certain regression coefficients, which can easily be computed adaptively in sensor nodes.

In the following we propose an approach to model the functions of problem (4), along with a strategy to achieve the optimal solution, namely the decision variables that minimize the cost function and satisfy the application requirements. Experimental based models of the reliability, delay and power consumption will be used.
5. EXPERIMENTAL BASED MODELS

We present the analysis of the dependency of the total power consumption, reliability and delay on the listening time and sleep time of the nodes. Simple experimental relations of the functions in problem (4) are derived so that the problem can be quickly solved by the nodes. The accurate analytical models of the reliability, delay, and power consumption of the duty-cycle algorithm with the random access control have been presented in [Fischione et al. 2009]. The analytical expressions are a function of listening time, sleep time, MAC parameters, and traffic load of a network. The drawback of these models is that they are highly non linear expressions that are difficult to use in practice. This motivates the experimental study of this paper.

The duty-cycle algorithm of IEEE 802.15.4 protocol was implemented on a test-bed using TelosB sensors [Polastre et al. 2005] running the Contiki operating system [Dunkels et al. 2004] based on the specifications of the IEEE 802.15.4 [IEEE 2006]. The implementation is available for download [Qin and Park 2011]. The values used for power consumption are those of the radio transceiver CC2420, which is featured by the TelosB. The length of preamble, ACK and data packets are 24, 11 and 56 bytes for data payload of 35 bytes respectively. $BE_{\text{min}} = 2$, $BE_{\text{max}} = 3$, and $NB_{\text{max}} = 2$ unless otherwise stated. The IEEE 802.15.4 defines one backoff as 20 symbols that correspond to 320 µs for 2.45 GHz. Since the hardware timer available for TelosB is based on a 32768 Hz clock, we use a backoff with duration of 305 µs instead of 320 µs. The current drawn is 18.8 mA in receive mode, 17.4 mA when transmitting at 0 dBm, 20 µA in idle mode and 1 µA in sleep mode. We consider a typical indoor environment with concrete walls. Each node is at a distance of around 5 m from the cluster-head.

We consider a star topology consisting of a number of nodes up to 12, and packet generation periods varying from 10 s to 60 s. We let $r$ be the average packet generation period by each node. Every node asynchronously generates a packet with probability $p$ for each slot time unit $S_b$ where $p = \frac{S_b}{r}$ and $S_b = 0.125$ s. The experimental based models are validated for different MAC parameters and transmission power. Linear regression is then used to compute the parameters of the experimental based models using the experimental results. We have chosen a linear regression because this allows us modeling the relations with quadratic functions and yields a closed form solution of the optimization problem (4).

5.1. Reliability Constraint

In this subsection, we provide an experimental based model for the reliability constraint (2) of problem (4), where we recall that the reliability is defined as the probability of successful packet reception. Figs. 3(a) and 3(b) show the reliability as obtained by the experiments as a function

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure4}
\caption{Reliability obtained by the experiments and the experimental based model given in Eq. (4) as a function of the sleep time with different data generation periods $\lambda = 10, 30, 60$ s for the number of transmitters $N = 8$.}
\end{figure}
Fig. 5. Residual sum of squares (RSS) of the reliability between the experimental results and the experimental based models given in Eq. (4) as a function of different data generation periods $\lambda = 10, 30, 60 \, \text{s}$ and number of nodes $N = 4, \ldots, 12$. The lower the RSS, the better the experimental based model.

Fig. 6. Residual sum of squares (RSS) of the reliability between the experimental results and the experimental based models given in Eq. (4) as a function of different MAC parameters $N_{B_{\text{max}}} = 1, \ldots, 5$ and transmission power level $TX = 0, -1, -3 \, \text{dBm}$. The lower RSS, the better the experimental based model.

of the listening time and sleep time with the data generation period $\lambda = 30 \, \text{s}$ and the number of transmitters $N = 8$, respectively. The vertical bars indicate the standard deviation as obtained out of 5 experimental runs of 30 min each.

Fig. 5(a) shows that reliability improves as the listening time increases. The improvement of the reliability however is negligible as the listening time increases above a certain value, i.e. $T_l \geq 6 \, \text{ms}$. The reason is that this listening time value is able to accommodate the total time spent for random backoff before sending a preamble at the transmitter and in handling hardware interrupts at the receiver most of the time when the traffic load is low. Fig. 3(b) on the other hand shows that the reliability decreases linearly as the sleep time increases for $T_l \geq 6 \, \text{ms}$. As the sleep time increases, the expected number of preambles increases, which increases contention during listening time.

From the observation of the dominant effect of sleep time on the reliability, we propose the following simple experimental based model for the reliability of problem (4):

$$R(T_s) \approx i_R + r_RT_s,$$ (4)

ACM Transactions on Sensor Networks, Vol. V, No. N, Article A, Publication date: January YYYY.
Duty-Cycle Optimization for IEEE 802.15.4 Wireless Sensor Networks

(a) Average delay as a function of the listening time with different sleep times.

(b) Average delay as a function of the sleep time with different listening times.

Fig. 7. Average delay obtained by the experiments as a function of the listening time $T_l = 3, \ldots, 10$ ms and sleep time $T_s = 0.1, \ldots, 2$ s with the data generation period $\lambda = 30$ s for the number of transmitters $N = 8$.

for $T_l \geq 6$ ms, where $i_R$ represents the intercept and $r_R$ denotes the slope of the line. The best value of $T_l$ will be determined to be 6 ms to minimize power consumption once similar observation is made for the delay in Section 5.2. We remark that the analytical model of the reliability proposed in [Fischione et al. 2009] validates the dominant effect of the sleep time on the reliability.

Fig. 4 shows the reliability as obtained by the experiments and the simple experimental based model given in Eq. (4) as a function of the sleep time $T_s = 0.1, \ldots, 2$ s with the data generation period $\lambda = 10, 30, 60$ s and the number of transmitters $N = 8$. The coefficients of Eq. (4) is computed by a simple linear regression.

The linear relation for reliability has been verified for various scenarios of the network with different network parameters such as traffic load, number of nodes, channel condition, network topology and MAC parameters based on the experiments. Fig. 5 shows the residual sum of squares (RSS) between the experimental results and the experimental based models given in Eq. (4) as a function of different number of nodes $N = 4, \ldots, 12$ and data generation periods $\lambda = 10, 30, 60$ s. The simple linear models for reliability are good approximations for different number of nodes and traffic load of the network. The RSS of the experimental based model given in Eq. (4) shows that the sleep time

ACM Transactions on Sensor Networks, Vol. V, No. N, Article A, Publication date: January YYYY.
is dominant parameter for the reliability. Furthermore, we remark that RSS increases as the traffic load increases due to the nonlinear factor for high contention. In a similar way, Fig. 6 reports RSS between the experimental results and the experimental based models given in Eq. (4) as a function of different MAC parameters $N B_{\text{max}} = 1, \ldots, 5$ and transmission power level $TX = 0, -1, -3 \text{ dBm}$. We observe that the experimental based model gives higher RSS for low transmission power due to the hidden node problem. These comparisons show that the reliability is well approximated by the linear relation given in Eq. (4) for the application we are concerned in this paper. The effect of listening time is negligible for the reliability compared to the effect of sleep time.

We use the experimental based model of the reliability to find the solution of problem (4) in Section 6. Now, we turn our attention to the delay constraint.

5.2. Delay Constraint

In this subsection, we provide an experimental based model for the delay constraint (3) of problem (4). Recall that the delay for a successfully transmitted packet is defined as the time interval from the instant the packet is generated, until the transmission is successful after receiving the corresponding ACK from the receiver. Figs. 7(a) and 7(b) show the average delay as obtained by the experiments as a function of the listening time and sleep time with the data generation period $\lambda = 30 \text{ s}$ and the number of transmitters $N = 8$, respectively. As the listening time decreases from $T_l = 6 \text{ ms}$, the delay increases as seen in Fig. 7(a). The reason is that the receiver frequently misses the preambles when the listening time is too short therefore the expected number of preambles to send a data packet so the delay increases. Once the listening time is large enough, most of the packets are received in the first listening time so the small value of the listening time compared to the sleep time results in negligible effect on the average delay for $T_l \geq 6 \text{ ms}$. On the other hand, we observe a good linear relationship between delay and sleep time. Based on this observation, we propose the following simple experimental based model for the average delay of problem (4):

$$D(T_s) \approx i_D + r_D T_s.$$  \hspace{1cm} (5)

for $T_l \geq 6 \text{ ms}$, where $i_D$ represents the intercept and $r_D$ denotes the slope of the line. This relationship is valid only when $T_s \geq T_l$. However, this is not a limitation, because, to save power, sensors have to use duty-cycles much smaller than 50%, which is compatible with $T_s \geq T_l$. The coefficients $i_D$ and $r_D$ are determined based on the experiments for different network parameters. A good linear relationship between the delay and sleep time is validated also through the analytical model of the delay proposed in [Fischione et al. 2009].
Fig. 9. Residual sum of squares (RSS) of the average delay between the experimental results and the experimental based models given in Eq. (5) as a function of different data generation periods $\lambda = 10, 30, 60$ s and number of nodes $N = 4, \ldots, 12$.

Fig. 10. Residual sum of squares (RSS) of the average delay between the experimental results and the experimental based models given in Eq. (5) as a function of different MAC parameters $NB_{\text{max}} = 1, \ldots, 5$ and transmission power level $TX = 0, -1, -3$ dBm. Recall that the lower RSS, the more accurate is the experimental based model.

Fig. 8 compares the average delay of the experimental results and the experimental based model given in Eq. (5) using linear regression as a function of the sleep time with different traffic generation periods $\lambda = 10, 30, 60$ s for the number of transmitters $N = 8$. The linear model given in Eq. (5) predicts well the experimental results.

In Figs. 9 and 10, we validate the experimental based models for delay given in Eq. (5) for different network parameters. Fig. 9 shows RSS between the experimental results and the experimental based models given in Eq. (5) as a function of different data generation periods $\lambda = 10, 30, 60$ s and number of nodes $N = 4, \ldots, 12$. We observe low values of RSS for various number of nodes and traffic loads. RSS increases as the traffic load and the number of nodes increase due to the nonlinear factor when the contention of the network increases. Similarly, Fig. 10 reports RSS between the experimental results and the experimental based models given in Eq. (5) as a function of different MAC parameters $NB_{\text{max}} = 1, \ldots, 5$ and transmission power level $TX = 0, -1, -3$ dBm. The effect of the listening time on the average delay is negligible similar to its effect on the reliability. We conclude that the delay is well approximated by the linear model given in Eq. (5) for the application we are concerned in this paper.
We will use the experimental based model of the average delay to find the solution of the optimization problem (4) in Section 6. Now, we investigate the power consumption.

5.3. Power Consumption

In this subsection, we provide an experimental based model for the average power consumption (1) of problem (4). We recall that the average power consumption is the sum of the expected power consumption to receive and send data packets. As we did for reliability and delay, it is possible to interpolate the power values obtained through the experiments as a function of the listening time and sleep time. Figs. 11(a) and 11(b) show the average power consumption as obtained by the experiments as a function of the listening time and sleep time with the data generation period $\lambda = 30$ s and the number of transmitters $N = 8$, respectively. We remind that our optimization problem is to minimize the power consumption while meeting the reliability and delay requirements in the packet transmission. Because the power consumption increases as the listening time increases for $T_l \geq 6$ ms in Fig. 11(a), it is natural to reduce the listening time by considering both reliability and delay performance. We set the listening time $T_l = 6$ ms because the reliability and delay sig-
Fig. 12. Average power consumption obtained by the experiments and the experimental based model given in Eqs. (6), (7), and (8) as a function of different sleep times and the packet generation periods $\lambda = 10, 30, 60$ s for the number of transmitters $N = 8$. 

ACM Transactions on Sensor Networks, Vol. V, No. N, Article A, Publication date: January YYYY.
nificantly degrade for $T_l < 6$ ms and the improvement of reliability and delay are negligible for $T_l > 6$ ms as observed in Figs. 3(a) and 7(a). In Fig. 11(b), we observe a tradeoff between the receiving cost of idle listening and transmission cost of preambles. While using a longer sleep time reduces the cost of idle listening at the receiver, it increases the transmission cost as the transmitter sends more preambles with possible contention. There is optimal value for the sleep time beyond which nodes waste more power in transmission than they save in reception.

In order to derive simple experimental based models, for a given $T_l$, we separate the average power consumption to receive and send data packets, $E_{rx}(T_s)$ and $E_{tx}(T_s)$ respectively. Such simple experimental based models for $E_{rx}(T_s)$ and $E_{tx}(T_s)$ and the average total power consumption, $E(T_s)$, result

$$E_{rx}(T_s) \approx i_{E_{rx}} + \frac{gE}{T_s}, \quad (6)$$

$$E_{tx}(T_s) \approx i_{E_{tx}} + r_ET_s, \quad (7)$$

$$E(T_s) \approx i_E + \frac{gE}{T_s} + r_ET_s. \quad (8)$$

where $i_E = i_{E_{rx}} + i_{E_{tx}}$ and the coefficients $i_{E_{rx}}, i_{E_{tx}}, r_E, gE$ are determined based on the experiments using linear regression. We remark that the analytical model of the average power consumption proposed in [Fischione et al. 2009] validates the quadratic relation between the average power consumption and sleep time.

Figs. 12(a) and 12(b) compare the average power consumption to receive and send data packets as obtained by the experiments and the experimental based model given in Eqs. (6) and (7), respectively, as a function of different sleep time and the data generation period $\lambda = 10, 30, 60$ s for the number of transmitters $N = 8$, respectively. The experimental based model for power consumption follows well the experimental results. In Fig. 12(c), we clearly observe a tradeoff between the receiving cost and transmission cost of longer sleep time. Therefore, it is critical to determine the optimal sleep time to balance the average power consumption to receive and send data packets.

Further analysis of the average power consumption reveals that the approximation given in Eq. (8) is good for various network scenarios. Fig. 13 shows RSS between the experimental results and the experimental based models given in Eq. (8) as a function of different data generation periods $\lambda = 10, 30, 60$ s and number of nodes $N = 4, \ldots, 12$. The RSS of the experimental based model given in Eq. (8) is small. Fig. 14 reports RSS between the experimental results and the experimental based models given in Eq. (8) as a function of different MAC parameters $NB_{\max} = 1, \ldots, 5$.
and transmission power level $TX = 0, -1, -3$ dBm. We observe that RSS increases as $NB_{\text{max}}$ increases due to the random backoff time of unslotted IEEE 802.15.4. These comparisons show that the average power consumption is well approximated by the model given in Eq. (8).

### 6. ADAPTIVE DISTRIBUTED ALGORITHM

In this section, we solve the optimization problem based on the experimental based models derived in Section 5. Furthermore, we describe the AODC algorithm to implement in practice the optimal solution.

As stated in Section 5, we set the listening time $T_l = 6$ ms and find the optimal value of sleep time. The reason is that the reliability and delay significantly degrade for $T_l < 6$ ms with negligible improvement for $T_l \geq 6$ ms as shown in Figs. 3 and 7 whereas the power consumption increases as the listening time increases.

By putting the experimental based models of the reliability and delay constraints and power consumption given in Eqs. (4), (5), and (8), respectively, it is possible to reformulate problem (4) by

$$\min_{T_s} \frac{i_E}{T_s} + \frac{g_E}{r_E} T_s + \frac{r_E T_s}{r_R} \quad \text{s.t.} \quad i_R + \frac{r_R T_s}{r_D} \geq R_{\text{min}},$$

$$i_D + \frac{r_D T_s}{r_D} \leq D_{\text{max}}.$$

This problem is quadratic because the cost function is quadratic and the constraints are in standard linear form. The optimal solution can then be expressed in closed form after using standard Lagrangian methods as follows [Boyd and Vandenberghe 2004]:

$$T_s^* = \min \left( \frac{g_E}{r_E}, \frac{R_{\text{min}} - i_R}{r_R}, \frac{D_{\text{max}} - i_D}{r_D} \right)$$

In Eq. (10), the first term is derived by taking the derivative of the cost function with respect to $T_s$ whereas the second and third terms are computed by using the reliability and delay constraints respectively. In the derivation of the equation, without loss generality, we assumed that the coefficient of the cost function $r_E > 0$, the coefficients of the reliability constraint $i_R > 0$, $r_R < 0$, and the coefficients of the delay constraint $i_D > 0$, $r_D > 0$. We propose the AODC algorithm described in Algorithm 1 at each receiver node. The main symbols of the algorithm and the default values of the algorithm’s parameters which we have used in the experiments are described in Table I. The goal of the algorithm is that a receiver node finds...
the optimal sleep time that minimizes power consumption for given reliability and delay constraints $R_{\min}$ and $D_{\max}$ based on the solution provided by Eq. (10). Meanwhile, the node periodically learns the coefficients of the mathematical models of Eqs. (4), (5), and (8) in an adaptive manner to the changes in the environment and network topology. The AODC algorithm therefore is composed of two phases: learning phase and optimization phase. The learning phase deals with the coefficient learning of the functions of the optimization problem, which is then solved in the optimization phase. The learning phase is needed to avoid recording in a look-up table the coefficients of the model for each possible configuration of the network. The size of the table to keep this information is not manageable. Moreover, for every receiver node of the network, it is usually not possible to know the exact configuration of the neighbors and their traffic. The learning phase of our algorithm does not require any explicit information about the traffic load and topology of the network thus minimizes the extra communication overhead throughout the network. We describe the running of the algorithm in the following.

The AODC algorithm requires that each receiver node estimates the reliability $\hat{R}$, delay $\hat{D}$ and power consumption $\hat{E}$ from the neighbors upon reception of each packet (line 6 of the algorithm). Then, the node periodically checks whether the desired reliability $R_{\min}$, delay $D_{\max}$ and power consumption $E^*$ values as requested by the application are achieved within a certain accuracy. If these estimated values $\hat{R}$, $\hat{D}$, and $\hat{E}$ are not met within certain factors (line 7 of the algorithm), then

**Algorithm 1: Pseudocode for the AODC algorithm.**

\begin{verbatim}
Input: $T_0$, $C_{\max}$, $\alpha_R$, $\alpha_D$, $\alpha_E$, $\delta$, $R_{\min}$, $D_{\max}$
Output: $T_s$
begin
    $t \leftarrow 0$;
    $T_s \leftarrow T_0$;
    $E^* \leftarrow -\infty$;
    forever do
        Update $\hat{R}$, $\hat{D}$, $\hat{E}$, $\hat{E}_{rx}$, $\hat{E}_{tx}$;
        if $\hat{R} < (1 - \alpha_R)R_{\min}$ or $\hat{D} > (1 + \alpha_D)D_{\max}$ or $\hat{E} > (1 + \alpha_E)E^*$ or $\hat{E} < (1 - \alpha_E)E^*$ then
            $C \leftarrow C + 1$;
            if $C > C_{\max}$ then
                // Learning phase.
                $\hat{R}_1 \leftarrow \hat{R}$, $\hat{D}_1 \leftarrow \hat{D}$, $\hat{E}_{rx,1} \leftarrow \hat{E}_{rx}$, $\hat{E}_{tx,1} \leftarrow \hat{E}_{tx}$;
                $T_s \leftarrow \delta T_s$;
                $\hat{R}_2 \leftarrow \hat{R}$, $\hat{D}_2 \leftarrow \hat{D}$, $\hat{E}_{rx,2} \leftarrow \hat{E}_{rx}$, $\hat{E}_{tx,2} \leftarrow \hat{E}_{tx}$;
                Update $\hat{r}_R$, $\hat{r}_D$, $\hat{r}_{E}$;
                $i_R(t) \leftarrow i_R$, $i_R(t) \leftarrow \hat{r}_R$;
                $i_D(t) \leftarrow i_D$, $i_D(t) \leftarrow \hat{r}_D$;
                $i_E(t) \leftarrow i_E$, $i_E(t) \leftarrow \hat{r}_E$;
                // Optimization phase.
                $T_s \leftarrow \min \left( \left( \frac{\hat{R}_{\min} - i_R(t)}{\hat{E}(t)} \right) \frac{R_{\min} - i_R(t)}{\hat{E}(t)} \right)$;
                $E^* \leftarrow \hat{E}(t) + \frac{\hat{E}(t)}{T_s}$;
            end
        else
            $C \leftarrow 0$;
        end
    end
end
\end{verbatim}
the learning phase is activated (lines 11-17 of the algorithm). Next we describe in detail how the estimation is performed.

The reliability is easy to estimate by using the sequence number of received data packets (line 6). To estimate the delay, each transmitter adds the delay between the packet generation time and the packet sending time to the payload of the data packet. However, this solution introduces an extra delay due to the limited speed of the serial peripheral interface (SPI) bus and the internal delay of the operating system. Note that before sending a packet, the microcontroller copies the packet data into the transmit buffer of the radio transceiver over the SPI bus. In [Osterlind and Dunkels 2008], the authors show that the packet copying is a critical issue when forwarding a packet. Hence, the transmitter adds the delay information of the previous data packet into the payload of the current data packet. Furthermore, by recording the transitions among transmit, receive, idle and sleep states, the transmitter is able to estimate its own average power consumption to receive and send packet packets. The transmitter then includes this information in the payload of the data packet.

When a receiver gets a data packet, it retrieves the packet delay and power consumption from the payload and estimates the reliability by tracking the sequence numbers for the corresponding neighbor. For the reliability and delay estimation, the receiver just finds the averages over the estimated values of each neighbor. For the power consumption on the other hand the receiver estimates its own average power consumption by recording its own state transitions and then averages together with the average power consumption of the neighbors. Because the number of measurements to estimate the reliability, delay and power consumption is small, the effect of measurement errors is critical for the accuracy of the experimental based model. We use the sliding window method to smooth the performance measurement for a given sleep time.

The condition to check if the reliability and average delay requirements are not met is specified as \( \tilde{R} < (1 - \alpha_R)R_{\min} \) and \( \tilde{D} > (1 + \alpha_D)D_{\max} \) respectively where \( 0 < \alpha_R < 1 \) and \( 0 < \alpha_D < 1 \) (line 7). The optimality of power consumption on the other hand is checked by \( \tilde{E} > (1 + \alpha_E)E^* \) and \( \tilde{E} < (1 - \alpha_E)E^* \), where recall that \( E^* \) is the expected optimal power consumption and \( 0 < \alpha_E < 1 \). \( E > (1 + \alpha_E)E^* \) can appear if new nodes enter the network or the link connectivity changes. On the other hand, \( E < (1 - \alpha_E)E^* \) can happen if nodes leave the network, hence the contention of random access mechanism and traffic load decreases, while meeting the requirement \( \tilde{R} > (1 - \alpha_R)R_{\min} \) and \( \tilde{D} < (1 + \alpha_D)D_{\max} \). In this case, since \( E^* \) is not the optimal value anymore, each node consumes more power than the actual optimal one even though the reliability and delay meet the application requirement. The constraints are relaxed by introducing the factors \( \alpha_R, \alpha_D, \) and \( \alpha_E \) to take into account the stochastic behavior. Each node keeps track of the number of times the requirements are not met (lines 8 – 9). If this number is greater than a threshold value, i.e. \( C > C_{\max} \), the node activates the learning phase (lines 11-17 of the algorithm), which we describe next.

In the learning phase, each node estimates the power consumption \( \tilde{E} \), reliability \( \tilde{R} \), and delay \( \tilde{D} \) for different sleep time \( T_s \), then runs simple linear regression to compute the coefficients of the experimental based model in Eqs. (4), (5), (8). In general, the linear regression gives better estimation as the number of the measurements increases. However, the higher the number of the measurements, the larger the memory requirement and computation load to run the linear regression.

Table I. Main symbols used in Algorithm 1.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_0 )</td>
<td>Initial sleep time</td>
</tr>
<tr>
<td>( C_{\max} )</td>
<td>Threshold of consecutive infeasible sets to activate the optimization algorithm, default 10</td>
</tr>
<tr>
<td>( \alpha_R )</td>
<td>Relaxation factor of ( R_{\min} ), default 0.9</td>
</tr>
<tr>
<td>( \alpha_D )</td>
<td>Relaxation factor of ( D_{\max} ), default 1.1</td>
</tr>
<tr>
<td>( \alpha_E )</td>
<td>Relaxation factor of ( E^* ), default 0.1</td>
</tr>
<tr>
<td>( \delta )</td>
<td>Update ratio of the sleep time, default 0.1</td>
</tr>
</tbody>
</table>

**ACM Transactions on Sensor Networks, Vol. V, No. N, Article A, Publication date: January YYYY.**
Therefore, each node uses the least number of measurements necessary to learn the coefficient of the experimental based model in each step.

When the learning phase starts, the node reduces the current optimal sleep time, which we denote by $T_{s_1}$, to $T_{s_2} = \delta T_{s_1}$ where $0 < \delta < 1$ (lines 11–13). By doing so, each node improves the reliability and delay while learning the change of the network environment. Then, the parameters of the experimental based model are computed (lines 14–17). For the reliability experimental based model given in Eq. (4), the estimated parameters of the intercept and slope are

$$
\hat{i}_R = \frac{1}{\delta - 1} \left( \delta \tilde{R}_1 - \tilde{R}_2 \right),
$$

$$
\hat{r}_R = \frac{1}{(\delta - 1) T_{s_1}} \left( \tilde{R}_2 - \tilde{R}_1 \right),
$$

(11)

where $\tilde{R}_1$ and $\tilde{R}_2$ are the estimated reliability corresponding to the sleep times $T_{s_1}$ and $T_{s_2}$ respectively. Note that the form of the linear regression of the reliability is same as the one used for the delay constraint and the average power consumption to send a data packet so equations similar to Eq. (11) allows us to estimate $\hat{i}_D, \hat{r}_D, \hat{r}_E$ and $\hat{i}_{E_{rx}}$. Similarly, we compute the coefficients of the experimental based model of the average power consumption to receive data packets given in Eq. (6) by estimating the average power consumption for two different sleep times $T_{s_1}, T_{s_2}$. The learned coefficients are

$$
\hat{i}_{E_{rx}} = \frac{\delta}{\delta - 1} \left( \frac{\tilde{E}_{rx,2} - \tilde{E}_{rx,1}}{\delta} \right),
$$

$$
\hat{g}_E = \frac{\delta T_{s_1}}{\delta - 1} \left( \frac{\tilde{E}_{rx,1} - \tilde{E}_{rx,2}}{\delta} \right),
$$

(12)

where $\tilde{E}_{rx,1}$ and $\tilde{E}_{rx,2}$ are the estimated average power consumptions to receive data packets corresponding to the sleep times $T_{s_1}$ and $T_{s_2}$ respectively. The sliding window is initialized to estimate the reliability, delay and power consumption. Otherwise, the convergence rate to estimate these parameters is very slow.

Once a receiver node learns the network environment by knowing the coefficients of the experimental based model of Eqs. (4), (5) and (8), the optimization phase of the algorithm starts (lines 18–19). The node sets the sleep time to its optimal value $T^*_s$ by using the solution derived in Eq. (10). If the problem is not feasible, it means that it is not possible to meet the reliability $R_{min}$ and delay requirements $D_{max}$ by tuning the sleep time. The application requirements must be relaxed so that feasibility is ensured and the problem can be solved.

The adaptive algorithm described so far assumed that all packet losses are due to the long sleep time. This assumption has allowed us to simplify the dependency of the reliability, delay and power consumption on the duty-cycle. However, in practice, links of IEEE 802.15.4 are bursty between bad and good delivery performance [Srinivasan et al. 2008]. If a node has a bad delivery link, reducing the sleep time does not improve the reliability and delay, but increases the power consumption. A node can avoid adjusting the parameters to a short burst length by keeping the length of the sliding window over which the averages for reliability, delay and power consumption are taken long enough compared to the burst length.

7. EXPERIMENTAL RESULTS

In this section, we analyze the performance of the AODC algorithm for tuning the duty-cycles under both stationary and transient conditions based on an extensive set of real-world experiments. In the stationary condition, the application requirements and network scenario are constant whereas they vary over time in the transient case. The experimental setup was described in Section 5. As we presented in Section 6, each receiver node estimates the reliability, delay, and power consumption of the network to run the AODC algorithm. The sequence number of the IEEE 802.15.4 MAC
header is used to estimate the reliability without extra overhead. Each transmitter adds the delay of the previous data packet into the payload of the current data packet. In addition, each node records the radio state transitions among transmit, receive, idle, and sleep state to estimate its own average power consumption to receive and send data packets and adds corresponding information into the payload. When a node receives ACK to the transmitted packet, it resets the number of state transitions regarding the power consumption. When each node receives a data packet, it first retrieves the information of sequence number, packet delay and power consumption of neighbors. Then, it computes the average reliability, delay and power consumption.

7.1. Protocol Behavior in Stationary Conditions
In this subsection, we analyze the performance metrics of the AODC algorithm under the stationary condition, namely without changing the application requirements (i.e., $R_{\text{min}}$ and $D_{\text{max}}$) and network scenarios.

First, we validate our optimization algorithm for different reliability and delay requirements. The optimal duty-cycle is obtained by using the AODC algorithm. Fig. 15 shows the reliability obtained by this algorithm with different reliability constraints $R_{\text{min}} = 0.9, 0.93, 0.96, 0.99$ and delay constraints $D_{\text{max}} = 0.05, 0.1, 1$ s for the number of transmitters $N = 8$.

Fig. 15. Reliability as a function of different delay requirements $D_{\text{max}} = 0.05, 0.1, 1$ s and reliability requirements $R_{\text{min}} = 0.9, 0.93, 0.96, 0.99$ for the number of transmitters $N = 8$.

Fig. 16. Average delay as a function of different reliability requirements $R_{\text{min}} = 0.8, 0.9, 0.95$ and delay requirements $D_{\text{max}} = 0.2, 0.4, 0.6, 0.8$ s for the number of transmitters $N = 8$. 

header is used to estimate the reliability without extra overhead. Each transmitter adds the delay of the previous data packet into the payload of the current data packet. In addition, each node records the radio state transitions among transmit, receive, idle, and sleep state to estimate its own average power consumption to receive and send data packets and adds corresponding information into the payload. When a node receives ACK to the transmitted packet, it resets the number of state transitions regarding the power consumption. When each node receives a data packet, it first retrieves the information of sequence number, packet delay and power consumption of neighbors. Then, it computes the average reliability, delay and power consumption.
Fig. 17. Comparison of the power consumption of X-MAC and AODC algorithm. AODC basically outperforms X-MAC, however the comparison is unfair is the sense that X-MAC is not explicitly designed for clustered topology, as it is AODC.

straints $D_{\text{max}} = 0.05, 0.1, 1$ s whereas Fig. 16 shows the average delay of the algorithm for different reliability requirements $R_{\text{min}} = 0.8, 0.9, 0.95$ and delay requirements $D_{\text{max}} = 0.2, 0.4, 0.6, 0.8$ s. As the delay requirement becomes more strict decreasing from $D_{\text{max}} = 1$ s to $D_{\text{max}} = 0.05$ s in Fig. 15, the reliability requirement $R_{\text{min}} = 0.9, 0.93$ is inactive. As observed in Fig. 16, the effect of both the reliability $R_{\text{min}} = 0.8, 0.9$ and delay requirements $D_{\text{max}} = 0.2, 0.4, 0.6, 0.8$ s for the average delay is negligible because the sleep time minimizing the power consumption is the dominant factor of the optimization problem. The average delay decreases as the reliability constraint becomes more strict $R_{\text{min}} = 0.95$ because the sleep time decreases to meet the reliability constraint.

Fig. 17 shows the power consumption obtained by X-MAC and AODC algorithm. Recall that X-MAC does not take into account random backoff, reliability and delay constraints. Therefore, for the sake of comparison of the AODC algorithm and X-MAC, we pose $R_{\text{min}} = 0$ and $D_{\text{max}} = \infty$, which implies neglecting the reliability and delay requirements, i.e., the power is minimized without constraints, as done in X-MAC. Our protocol outperforms X-MAC in all the scenarios considered. Specifically, when the packet generation period is high (300 s) the difference is small, but as the packet generation period decreases the improvement is substantial. The main reason for this difference is that the nodes consume much less power in packet transmission compared to the model in [Buettnet et al. 2006]. X-MAC is based on the assumption that the transmitter sends preamble packets back to back until the receiver wakes up while actually there is random backoff before packet transmissions during which the transmitter puts its radio in sleep mode. Since the transmit power dominates the receive power much earlier according to the model in [Buettnet et al. 2006], the optimal listening time becomes considerably higher compared to the actual optimal listening time.

7.2. Protocol Behavior in Transient Conditions

The performance analysis carried out so far assumed that the number of nodes and traffic configuration are fixed. This assumption has allowed us to verify the effectiveness of the AODC algorithm for IEEE 802.15.4 in steady state conditions. However, one of the critical issues in the design of wireless networks is time varying condition. Therefore, in the following analysis, we will investigate the performance of the AODC algorithm when the number of nodes and traffic load changes over time.

We now compare our AODC algorithm to the algorithm AADCC proposed in [Merlin and Heinzelman 2010]. AADCC employs a simple linear increase/linear decrease of the sleep time, where whenever five consecutive packets are successfully sent to the destination, the sleep time is increased by 0.1 s. Otherwise, each node decreases the sleep time by 0.25 s. We consider AADCC due to its implementation simplicity with respect to the much more complex DDCC algorithm also proposed in [Merlin and Heinzelman 2010]. We remark that AADCC considers only the reliabil-
Fig. 18. Robustness of AODC algorithm to the changes in the number of nodes: sleep time, reliability and delay when the number of nodes changes sharply from $N = 10$ to $N = 15$ at time 300 s. The packet generation period is $\lambda = 10$ s and the reliability and delay constraints are $R_{\text{min}} = 0.9$ and $D_{\text{max}} = 0.5$ s respectively. Note that “AADCC” refers to the adaptive algorithm in [Merlin and Heinzelman 2010].

Fig. 19. Robustness of AODC algorithm to the changes in the traffic load: sleep time, reliability and delay when the traffic load changes sharply from $\lambda = 30$ s to $\lambda = 10$ s at time 300 s. The number of transmitters is $N = 10$, the reliability and delay constraints are $R_{\text{min}} = 0.8$ and $D_{\text{max}} = 3$ s respectively.

ity while the AODC algorithm controls both reliability and delay of the network. Hence, AADCC does not support different reliability and delay requirements of applications while our algorithm is adaptive to them as shown in Figs. 15 and 16.

Fig. 18 shows the variations in sleep time, reliability and packet delay of the adaptive algorithm proposed in this paper and AADCC when the number of nodes changes from $N = 10$ to $N = 15$. At time 300 s, the number of nodes suddenly increases to 15 whereas the experimental based model in use is still the one for $N = 10$. This causes a significant decrease of the reliability due to the high contention level as shown in the Fig. 18(b). The sleep time is updated by starting the learning phase of our adaptive algorithm. When the receiver node detects the change of network condition
due to greater number of nodes, it initializes the sliding window and decreases the sleep time to $\delta T_s$ where $\delta = 0.1$. After changing the sleep time, the node measures the reliability and delay of the network. The node then runs the optimization phase of the AODC algorithm and updates the sleep time to 0.102 s. We observe that the convergence of the sleep time of the AODC algorithm is very fast without significant oscillations. Also, we observe the high correlation between the sleep time and packet delay. By contrast, the sleep time of AADCC oscillates between 0 s and 0.65 s instead of converging, which is not desirable. Note that the sleep time of AADCC is zero at some points in time due to its simple linear increase/linear decrease mechanism. Although the reliability of the AODC and AADCC are similar, the delay of AADCC has a very high variance.

Fig. 19 presents the behavior of AODC and AADCC when the traffic load changes suddenly from $\lambda = 30$ s to $\lambda = 10$ s at time 300 s. The experimental based model estimated by the AODC algorithm for $\lambda = 30$ s needs to be changed once the traffic load changes. Similar to the case where the number of the nodes changes, the node runs the learning phase of the AODC algorithm since the measured reliability does not meet the reliability requirement due to the high traffic load. The algorithm then finds the new optimal sleep time during the optimization phase. Fig. 19(a) shows that the node updates the sleep time from 2.34 s to 0.27 s due to the poor reliability after the traffic load changes at time 300 s. The figure indicates that the system reacts correctly to the changes of traffic configuration after updating the experimental based model in few seconds. After the sleep time is optimized, the average delay converges to around 0.18 s. We observe that the packet delay is about five times lower than the one measured before time 300 s in Fig. 19(c). Specifically, we have a reduction in the average delay and a shorter tail for the delay distribution after changing the sleep time. The reliability requirement $R_{\text{min}} = 0.9$ is fulfilled with some fluctuations after the traffic load increases. The sleep time of AADCC oscillates between 0 s and 2.4 s without converging. Although the reliability of AADCC is higher than the adaptive algorithm, it consumes more power. Furthermore, AADCC does not have control of the delay. Recall that our target is to meet the reliability and delay requirements rather than just improving the reliability or delay performance.

8. CONCLUSIONS

We presented the AODC algorithm to minimize the power consumption while guaranteeing reliability and delay requirements of the application for the IEEE unslotted 802.15.4 sensor networks. This approach represents a major advancement with respect to existing solutions, such as X-MAC and AADCC protocols, because the parameters of the underlying model are able to gracefully adapt to the variations in the application requirements and network topology. The AODC algorithm is easily implementable on top of random access mechanism of unslotted IEEE 802.15.4 standard. Simple experimental based models are used to derive the cost function and constraints of the optimization problem as a function of the sleep time. This simplification allows to solve the optimization problem in closed form, hence making it possible to compute the optimal solution at the sensor nodes. The learning phase of the experimental based model is proposed to adaptively react to the changes in the network. We provided a test-bed implementation of the protocol with TelosB sensors and Contiki OS. Furthermore, we investigated the performance of the AODC algorithm under both stationary and transient conditions by experiments. Experimental results showed that the AODC algorithm is efficient and ensures a longer lifetime of the network. We showed that, even if the number of active nodes and traffic configuration suddenly change, AODC algorithm allows the network to adapt quickly and operate at the optimal parameter by continuously learning the experimental based models.

REFERENCES


Duty-Cycle Optimization for IEEE 802.15.4 Wireless Sensor Networks


