

# Towards Energy-Fairness in Asynchronous Duty-Cycling Sensor Networks

Zhenjiang Li<sup>1</sup>, Mo Li<sup>1</sup>, and Yunhao Liu<sup>2</sup>

<sup>1</sup>School of Computer Engineering, Nanyang Technological University

<sup>2</sup>MOE Key Lab for Information System Security, School of Software,  
Tsinghua National Lab for Information Science and Technology, Tsinghua University

Email: lzjiang@cse.ust.hk, limo@ntu.edu.sg, liu@cse.ust.hk

**Abstract**—In this paper, we investigate the problem of controlling node sleep intervals so as to achieve the min-max energy fairness in asynchronous duty-cycling sensor networks. We propose a mathematical model to describe the energy efficiency of such networks and observe that traditional sleep interval setting strategy, i.e., operating sensor nodes with identical sleep intervals, or intuitive control heuristics, i.e., greedily increasing sleep intervals of sensor nodes with high energy consumption rates, hardly perform well in practice. There is an urgent need to develop an efficient sleep interval control strategy for achieving fair and high energy efficiency. To this end, we theoretically formulate the Sleep Interval Control (SIC) problem and find it a convex optimization problem. By utilizing the convex property, we decompose the original problem and propose a distributed algorithm, called GDSIC. In GDSIC, sensor nodes can tune sleep intervals through a local information exchange such that the maximum energy consumption rate in the network approaches to be minimized. The algorithm is self-adjustable to the traffic load variance and is able to serve as a unified framework for a variety of asynchronous duty-cycling MAC protocols. We implement our approach in a prototype system and test its feasibility and applicability on a 50-node testbed. We further conduct extensive trace-driven simulations to examine the efficiency and scalability of our algorithm with various settings.

## I. INTRODUCTION

Recent several years have witnessed the great success of Wireless Sensor Networks (WSNs). As a promising technique, WSNs have spawned a variety of critical applications in practice. In WSNs, sensor nodes are usually powered by batteries, while frequent replacements of such power sources are normally prohibited. To close the gap between limited energy supplies of sensor nodes and the long-term deployment requirement in many applications, recent research works suggest to operate sensor nodes in a *duty-cycling* work mode [1]. In duty-cycling WSNs, radios of sensor nodes are controlled on a periodical basis, alternating between active and dormant states. In the active state, sensor nodes can send or receive data, while in the dormant state they switch radios off to save energy. A typical application that we have envisioned is a project aiming at monitoring and analyzing the raw water quality in Singapore. The maintenance and manpower costs are high for examining the sensor nodes and replacing the batteries in the project. Hence, we apply the duty-cycling technique to address such a problem. For instance, with a 5%

duty cycle, sensor nodes have radios on only 5% of the time. The duty-cycling work mode thus significantly reduces energy consumption rates of sensor nodes and dramatically prolongs the network lifetime.

The duty-cycling operation has been employed in a variety of MAC-layer protocols, which can be basically classified into *synchronous* and *asynchronous* two categories. Typical synchronous protocols, like [1]–[3], enable sensor nodes to synchronously sleep and wake up, providing intermittent network services. The required time synchronization introduces tremendous communication overhead and computation complexity. Asynchronous protocols, however, allow sensor nodes to operate independently. At an arbitrary time instance, a subset of sensor nodes operate to provide consistent network services. Most asynchronous protocols typically employ *Low Power Listening (LPL)* based approaches [4]–[6], including the original LPL technique or some other optimized techniques like *strobed preamble*, to achieve asynchronous data transmissions. The basic principle of those protocols is that prior to the data transmission, a sender transmits a preamble lasting as long as the sleep period (i.e. *sleep interval*) of the receiver, the receiver is, thus, guaranteed to detect the preamble and receive data. Compared with synchronous protocols, asynchronous protocols are free of time synchronization and robust to network dynamics, which are beneficial for large-scale deployments. Recently, some variant techniques, e.g., *Low Power Probing (LPP)*, have been proposed to enable receiver-initiated duty-cycling data transmissions. As all those above techniques share similar energy efficiencies, for the sake of clear presentation, we take LPL-based approaches as a vehicle to discuss the energy fairness issue in asynchronous duty-cycling sensor networks, and further extend our analysis and solution to other variant techniques.

Though the asynchronous duty-cycling operation releases the constraint of time synchronization and enables robust sensor networks in dynamic environments [7], [8], there remain excessive challenges for applying such an operation to manage limited energy supplies of sensor nodes and approach a long network lifetime. First, the choice of sleep interval at any given node affects not only its own energy drain to periodically access the channel, but also the energy consumption

of neighbor nodes communicating with it. In particular, by selecting a relatively large sleep interval, one sensor node will poll the channel less frequently with reduced energy drain. On the other hand, as LPL requires that preambles sent from senders should cover the entire sleep periods of receivers, setting a large sleep interval unavoidably increases the energy consumption of packet senders for the current recipient node. Such an energy trade-off challenges the appropriate choice of sleep intervals for different sensor nodes, and we call the problem *Sleep Interval Control (SIC)*. Second, the traffic load usually distributes unevenly and varies in the network in many applications. As the traffic load directly affects the preamble and wake-up time of individual sensors as well, the choice of sleep intervals cannot be determined separately from the traffic load awareness. If the SIC strategy is not well designed, certain nodes could rapidly deplete their energy and become the energy bottleneck, which severely breaks the network-wide energy fairness and thereby shortens the network lifetime. Thus, SIC becomes more challenging as it should be traffic-aware. In addition, the problem will get even worse if the network scale is large, demanding distributed solutions.

There have been excessive studies tailored for achieving the energy fairness to prolong the network lifetime of sensor networks. Nevertheless, they cannot be directly applied to the asynchronous duty-cycling context [9]–[12]. There have also been attempts made towards the SIC problem in duty-cycling WSNs. Most of them, however, investigate bounding the end-to-end transmission delay or adjusting the energy consumption of sensor nodes in a heuristic fashion and ignoring the traffic impact [13]–[16]. None of them tackles the SIC problem with a general setting to prolong the network lifetime in a distributed manner. So far as we know, many fundamental issues in SIC are not well understood and an instrument to tackle such problems is still lacking to the community.

In this paper, we thoroughly investigate the SIC problem to achieve min-max energy fairness in asynchronous duty-cycling sensor networks. The contributions of this paper are as follows. We propose a mathematical model to describe energy efficiency of sensor nodes in existing LPL based asynchronous duty-cycling sensor networks, which captures the essential energy trade-off between senders and receivers. Based on the proposed model, we observe that existing simple sleep interval control mechanisms perform far from optimal, and there is an urgent need to develop an efficient SIC strategy. Aiming at dealing with the SIC problem in general, we theoretically formulate such a problem and find it a convex optimization problem. Based on the convex property, we decompose the original problem into sub-optimization problems, and develop a distributed algorithm, called GDSIC, to approach the optimal result. In GDSIC, with a solely local information exchange, sensor nodes can determine how to adjust their sleep intervals such that all sensor nodes within the network converge to optimal sleep interval settings and the maximum

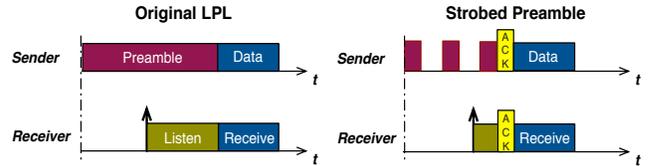


Fig. 1. Illustration of the LPL Technique

energy consumption rate in the network can be minimized. The GDSIC algorithm is self-adjustable to the traffic load variance and is able to serve as a unified framework for a variety of underlying asynchronous duty-cycling protocols. We implement a prototype system on a 50 TelosB Mote testbed. The experiment results validate the feasibility and applicability of the proposed approach in practice. We further conduct extensive and large-scale trace-driven simulations to examine the efficiency and scalability of the proposed algorithm.

The rest of this paper is organized as follows: related works are reviewed in Section II. In Section III, we model the energy efficiency of sensor nodes and evaluate traditional SIC strategies. We formulate the SIC problem and propose our solution in Section IV. In Sections V and VI, we examine the performance of our approach. We conclude in Section VII.

## II. RELATED WORK

In existing literatures, the first reported asynchronous MAC-layer protocol is B-MAC [4], which applies the original LPL technique. Afterward, subsequent protocols, like X-MAC [5], C-MAC [6], WiseMAC [17], and PW-MAC [18] are essentially similar to B-MAC. However, some optimizations, including the strobed preamble and predictive wake-up techniques, have been introduced in those protocols to further reduce the energy consumption. Since preambles are sent from senders, aforementioned asynchronous protocols are also referred to as *sender-initiated* protocols. Different from sender-initiated protocols, recently, some *receiver-initiated* protocols have been proposed, such as RI-MAC [19], A-MAC [20], etc, which are mainly designed to avoid collisions and unify services, by employing the LPP technique. Based on [16], energy drains in receiver-initiated protocols can be similarly analyzed as sender-initiated ones. In this paper, we take the LPL-based protocols as an instrumental vehicle due to LPL's availability in the standard TinyOS distribution, while dealing with LPP-based protocols as a promising extension.

There are also some primary efforts to control sleep intervals in WSNs. [13] proposes Dutycon to achieve a dynamic duty cycle control for end-to-end delay guarantee. In both [14] and [15], multi-objective optimization formulations are introduced, covering transmission reliability, end-to-end delay, and energy consumption. Optimization problems are solved by classical methods in a centralized manner. IDEA in [16] integrates multiple networking services, like LPL adjustment, energy-aware routing, and localization. Sensor nodes balance the

local energy consumption in a heuristic fashion and it is not clear how close the achieved performance is from the optimal result. In addition, the impact of the traffic load is ignored in [16]. As energy is the most significant issue limiting the network performance [21], different from previous works, we focus on controlling sleep intervals to achieve a min-max energy fairness so that the network lifetime can be notably prolonged. To make our approach practical, we require that the solution should be completely distributed and self-adjustable to the traffic variance, which is common in many applications, and serves as a unified framework applicable to a variety of existing asynchronous MAC-layer protocols. So far as we know, such an instrument is still lacking.

### III. PROBLEM SPECIFICATION AND DESIGN CHALLENGES

In this section, we mathematically characterize the energy efficiency<sup>1</sup> of sensor nodes with LPL-based asynchronous protocols, and evaluate existing sleep interval setting and control strategies in practice.

As depicted in Fig. 1 (left), a sender transmits a *long* preamble prior to the data transmission with the original LPL. After the receiver wakes up and detects the preamble, it keeps awake to receive data. Later, such a working mechanism has been further optimized due to the low energy efficiency at the receiver side, and the most representative example is the *strobed preamble* technique. As shown in Fig. 1 (right), instead of sending a long preamble, a series of short preambles are sent such that intended data can be transmitted without waiting until the end of the long preamble. Since such a technique notably increases the energy efficiency and robust to dynamic environments, it has been widely used in large-scale WSNs in practice, like GreenOrbs [22], and released as the default LPL-based MAC protocol in TinyOS. As optimized techniques are proposed based on the original LPL design, we first investigate energy consumption rates of sensor nodes with the original LPL technique in this section, then we observe that later proposed protocols can be unified as its special cases. Before we proceed, for any sensor node (e.g.  $i$ ) in the network, we introduce two notations:

- $r_i$  is the overall energy consumption rate of node  $i$ .
- $T_i^{slp}$  is the sleep interval of node  $i$ .

In general, each  $r_i$  is the summation of energy consumption rates for packet transmitting ( $r_i^{tx}$ ), packet receiving ( $r_i^{rc}$ ), channel polling ( $r_i^{cp}$ ) and overhearing ( $r_i^{oh}$ ) at sensor node  $i$ . As a result,  $r_i$  can be characterized by:

$$r_i = r_i^{tx} + r_i^{rc} + r_i^{cp} + r_i^{oh}. \quad (1)$$

After specifying each term in Eq. (1), we obtain a general expression for the overall energy consumption rate of any sensor node  $i$  in the following theorem:

<sup>1</sup>Without loss of generality, we focus on the rate of energy consumption (i.e., the energy drain in one unit time) in this section, as the total energy consumption can be obtained by multiplying the rate and the time duration.

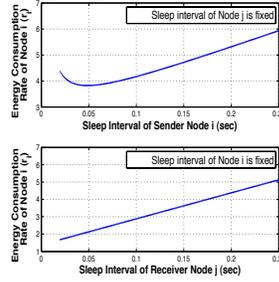


Fig. 2.  $r_i$  vs. Sleep intervals with original LPL technique

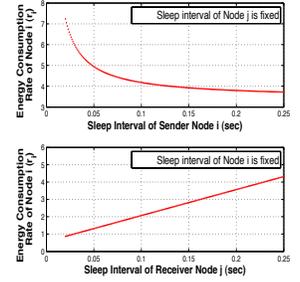


Fig. 3.  $r_i$  vs. Sleep intervals with strobed preamble technique

*Theorem 1: With the LPL technique, the overall energy consumption rate at any sensor node  $i$  can be unified by*

$$r_i = \lambda_i \cdot T_j^{slp} + \frac{\gamma_i}{T_i^{slp}} + \zeta_i \cdot T_i^{slp} + \tau_i, \quad (2)$$

where node  $j$  receives the packets sent from node  $i$ ,  $\lambda_i$ ,  $\gamma_i$ ,  $\zeta_i$  and  $\tau_i$  are coefficients to simplify the expression of  $r_i$ .

The detailed derivation of Theorem 1 can be found in our technical report [23]. Based on Theorem 1, we will 1) evaluate existing sleep interval setting and control strategies in practice; and 2) identify design challenges for the SIC problem.

#### A. Problem Specifications

To the best of our knowledge, most deployed WSNs in practice employ the identical sleep interval setting due to the design and implementation simplicities. However, it is well known that the in-network traffic load is usually unevenly distributed [24], [25] and recent measurement studies, like [7], have also reported such a phenomenon. We observe that such a simple strategy may lead to heterogenous energy drains and hardly achieve the energy fairness in the network. As a result, the network lifetime will be severely limited.

*Theorem 2: The identical sleep interval setting usually results in heterogenous energy consumption rates in practice.*

The rigorous interpretation for Theorem 2 can be found in our technical report [23], while we briefly explain Theorem 2 here. According to Theorem 1, we can demonstrate that the energy consumption rate  $r_i$  of any sensor node  $i$  is mainly determined by its outgoing (transmitting) traffic rate  $f_i^{tx}$  when all sensor nodes are set an identical sleep interval. As aforementioned, the network traffic in practice is normally heterogenous. Therefore, sensor nodes in heavy traffic regions are prone to suffer more frequent preamble time and longer data receiving time. As a consequence, those sensor nodes tend to run out of energy first, and traffic loads are prone to dominate the lifetime of sensor nodes when all sleep intervals are set to be equal.

Theorem 2 essentially demonstrates that due to the inherent *uneven* nature of traffic loads in practice, the widely adopted sleep interval setting policy in existing sensor networks fails to gain a good performance in terms of the energy efficiency. To

deal with such an issue, sleep intervals should be controlled dynamically with respect to sensors' energy draining speeds and traffic load variances. An intuitive solution is to increase the sleep interval of a sensor node greedily if its energy consumption rate becomes higher [16]. The rationale behind is that prolonging the sleep interval of this sensor node compensates its fast energy consumption. However, as we will show in Theorem 3, the hardness of the SIC problem is beyond such an intuition.

*Theorem 3: The greedy SIC strategy by increasing sleep intervals of sensor nodes with large energy consumption rates hardly achieve the min-max energy fairness in WSNs.*

In Fig. 2 and Fig. 3 we show the energy consumption rate of a sender node with respect to sleep interval settings of itself and its receiver. The upper figure in either Fig. 2 or Fig. 3 depicts that for any sensor node  $i$ , how does  $r_i$  in Eq. (2) vary with  $T_i^{slp}$  when  $T_j^{slp}$  is fixed. If we focus on each individual sensor node, its energy consumption rate is indeed decreased in some scenarios when the sleep interval increases. As depicted in Fig. 3 (upper), the strobed preamble technique belongs to this category. However, there exist sufficient exceptions. For instance, sensor nodes adopting the original LPL technique in the region with high traffic loads as shown in Fig. 2 (upper). It hinders above intuitive heuristics to be applied directly in general.

On the other hand, from the network perspective, Eq. (2) implies that after a sensor node increases its sleep interval, energy consumption rates of its senders increase accordingly. The lower figure in either Fig. 2 or Fig. 3 depicts how does  $r_i$  in Eq. (2) vary with  $T_j^{slp}$  when  $T_i^{slp}$  is fixed. As a matter of fact, the sleep interval adjustment of one sensor node will trigger senders to tune their own sleep intervals as well. In the greedy strategy, energy drains of sensor nodes are balanced within neighborhoods, which essentially follows the "water-leveling" mechanism. By doing so, energy consumption rates of sensor nodes could be converged to a compromised value. On the other hand, the initial sleep interval setting has implicitly defined an interval, within which the min-max energy fairness can be adjusted by the greedy strategy. However, there is no guarantee that the optimal min-max energy fairness falls within the formed interval exactly. Therefore, the greedy strategy is not always effective, which challenges the algorithm design for SIC. In the next subsection, we will specify the design challenges for the SIC problem in asynchronous duty-cycling sensor networks.

### B. Design Challenges

Based on above discussions, we can summarize the design challenges for the SIC problem as follows:

- Increasing sleep interval of one sensor node does not necessarily reduce its own energy consumption rate.
- A sensor node increases its own sleep interval to save energy; nevertheless, energy consumption rates of the packet senders of the current receiver may increase.

- The achieved energy fairness may be far away from the optimal result if the sleep interval is not carefully controlled.

In the next section, we will introduce our solution to deal with those challenges to achieve an optimized sleep interval control.

## IV. PROBLEM FORMULATION AND ALGORITHM DESIGN

The sensor network is modeled as an undirected graph  $G = \{V, E\}$ , where  $V$  and  $E$  represent the sets of sensor nodes and wireless links, respectively. According to Theorem 1, the energy consumption rate of an arbitrary sensor node  $i$  in the network can be expressed as  $r_i = \lambda_i \cdot T_j^{slp} + \gamma_i / T_i^{slp} + \zeta_i \cdot T_i^{slp} + \tau_i$ , where  $j$  is the receiver<sup>2</sup> of node  $i$ . To control the energy consumption rate in the network, we introduce a set of variables  $R_i$ s and require that  $\lambda_i \cdot T_j^{slp} + \gamma_i / T_i^{slp} + \zeta_i \cdot T_i^{slp} + \tau_i \leq R_i$  for each  $i$ . As previously mentioned, by determining an appropriate sleep interval for each sensor node, the Sleep Interval Control (SIC) problem aims at minimizing the maximum energy consumption rate (i.e., the min-max energy fairness) in the network to prolong the network lifetime, which can be captured by the model from Eq. (3) to Eq. (5) as follows:

$$\min \quad \max_i \{R_i\} \quad (3)$$

$$s.t. \quad \lambda_i \cdot T_j^{slp} + \frac{\gamma_i}{T_i^{slp}} + \zeta_i \cdot T_i^{slp} + \tau_i \leq R_i, (i, j) \in E, \quad (4)$$

$$0 < T_i^{slp}, i \in V. \quad (5)$$

Constraint (4) specifies that the energy consumption rate of each sensor node is bounded from above by the variable  $R_i$ . Constraint (5) guarantees that sleep intervals have positive values. The coefficients  $\lambda_i$ ,  $\gamma_i$ ,  $\zeta_i$ , and  $\tau_i$   $i \in V$ , are all positive as well. Thus, constraints (4) and (5) implicitly ensure that  $R_i > 0$ . In the end, the objective function (3) minimizes the maximum  $R_i$  so that the global min-max energy fairness can be achieved in the network.

A straightforward way to obtain the optimal SIC result based on above formulation is as follows:

- Each sensor  $i$  measures its own traffic load, calculates  $\lambda_i$ ,  $\gamma_i$ ,  $\zeta_i$ , and  $\tau_i$ , and reports the calculated coefficients to a central information collector, e.g. the sink.
- Based on the harvested information from each sensor node, the sink globally solves Eqs. (3) to (5).
- The sink node disseminates the optimal sleep interval setting to the entire network.
- To be traffic variance-aware, above three steps are repeated periodically or triggered via the sink when traffic dynamics are detected.

<sup>2</sup>At the current stage, we focus on the case, in which sensor node  $i$  has one receiver only. Such a scenario is common in practice and it can be found when packets are transmitted following a tree structure, e.g. CTP [26]. However, our proposal is not limited to the tree structure and we will discuss the multi-receiver case in subsection IV-C.

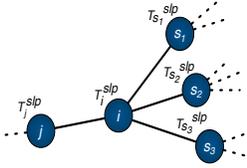


Fig. 4. Local structure for sensor  $i$

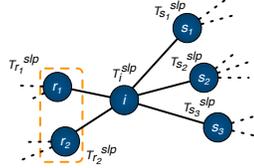


Fig. 5. Multi-receiver scenario

However, such a centralized solution normally incurs tremendous communication overhead and complicated cooperation among sensor nodes, which hinders the scalability and applicability of the solution for large-scale WSNs. To overcome those limitations, we now introduce a distributed approach to perform sleep interval control at individual sensor nodes.

#### A. Distributed Sleep Interval Control Problem

We decompose the original SIC problem to each sensor node and focus on a local structure of an arbitrary sensor node  $i$  in the network. As depicted in Fig. 4, node  $j$  is the receiver of sensor  $i$  and node  $s_k$ ,  $k = 1, 2, \dots, K$ , is a sender of sensor  $i$ , where  $K$  is the total number of potential senders. By exchanging information with those neighboring nodes, sensor node  $i$  can determine its local-optimal sleep interval based on the formulation from Eq. (6) to Eq. (9).

$$\min R_i, \quad (6)$$

$$\text{s.t.} \quad \lambda_i \cdot T_j^{slp} + \frac{\gamma_i}{T_i^{slp}} + \zeta_i \cdot T_i^{slp} + \tau_i \leq R_i, (i, j) \in E \quad (7)$$

$$\lambda_k \cdot T_i^{slp} + \frac{\gamma_k}{T_k^{slp}} + \zeta_k \cdot T_k^{slp} + \tau_k \leq R_i, (k, i) \in E \quad (8)$$

$$0 < T_i^{slp}, i \in V. \quad (9)$$

As  $T_i^{slp}$  affects energy consumption rates of both node  $i$  and its senders in Eqs. (7) and (8), the variable  $R_i$  bounds energy drains in the local region from above to control the energy trade-off between  $i$  and each sender  $s_k$ . Similar to the original SIC problem,  $R_i$  in the objective function (6) is minimized to obtain a local min-max energy fairness. We denote Eqs. (6) to (9) as the Distributed SIC (D-SIC) problem. The following lemma reveals the essence of both SIC and D-SIC problems.

*Lemma 1: The SIC problem and the D-SIC problem are both convex optimization problems.*

Conclusions made by Lemma 1 are clear as all constraints and objective functions in both SIC and D-SIC problems are convex. Due to the page limitation, we skip the detailed proof. In the D-SIC problem, the total amount of constraints is bounded by the number of senders of sensor node  $i$ . According to [7], each sensor node only needs to solve one local D-SIC problem with a small number of constraints (e.g.  $< 8$ ). As a result, a variety of mature and lightweight techniques can be adopted in practice, such as the interior-point method [27], in which the optimal result can be found within guaranteed iterations. Even D-SIC problems can be solved locally, there remains one critical issue not answered yet: how to ensure that

such distributed computations eventually lead to the global optimal result? The answer will be given when we introduce the Distributed SIC (DSIC) algorithm in the next subsection.

Before we proceed, we particularly investigate the D-SIC problem for a set of asynchronous protocols based on LPL with the strobed preamble technique, including X-MAC, C-MAC, and so on, because its aforementioned significance in practice. Due to the special properties, sensor nodes with this type of protocols can avoid using iterative algorithms to solve their own D-SIC problems; instead, close-form expressions can be obtained to further simplify the system design. Eqs. (7) and (8) with the strobed preamble technique in the D-SIC problem can be replaced by Eqs. (10) and (11), respectively.

$$\lambda_i \cdot T_j^{slp} + \frac{\gamma_i}{T_i^{slp}} + \tau_i \leq R_i, (i, j) \in E \quad (10)$$

$$\lambda_k \cdot T_i^{slp} + \frac{\gamma_k}{T_k^{slp}} + \tau_k \leq R_i, (k, i) \in E \quad (11)$$

As  $\gamma_i$  and  $T_i^{slp}$  are both positive, Eq. (10) implies  $R_i > \lambda_i \cdot T_j^{slp} + \tau_i$ , which further yields:  $T_i^{slp} \geq \gamma_i / (R_i - \lambda_i \cdot T_j^{slp} - \tau_i)$ . On the other hand, since  $\lambda_k > 0$ , based on Eq. (11), we can further obtain  $T_i^{slp} \leq (R_i - \gamma_k / T_k^{slp} - \tau_k) / \lambda_k$ . Then, for each sender  $k$ , we have:

$$\frac{\gamma_i}{R_i - \lambda_i \cdot T_j^{slp} - \tau_i} \leq T_i^{slp} \leq \frac{R_i - \gamma_k / T_k^{slp} - \tau_k}{\lambda_k},$$

$$\Rightarrow R_i^2 - \phi_i^{j,k} \cdot R_i + \omega_i^{j,k} \geq 0, \quad (12)$$

$$\Rightarrow \bar{R}_i = \max_k \{ (\phi_i^{j,k} + \sqrt{(\phi_i^{j,k})^2 - 4\omega_i^{j,k}}) / 2 \}, \quad (13)$$

where  $\bar{R}_i$  indicates the selected upper bound for energy consumption rates in the local region,  $\phi_i^{j,k} \triangleq \lambda_i \cdot T_j^{slp} + \tau_i + \gamma_k / T_k^{slp} + \tau_k$ , and  $\omega_i^{j,k} \triangleq (\lambda_i \cdot T_j^{slp} + \tau_i)(\gamma_k / T_k^{slp} + \tau_k) - \gamma_i \cdot \lambda_k$ . Note that  $(\phi_i^{j,k})^2 - 4\omega_i^{j,k}$  can be transformed to  $(\lambda_i \cdot T_j^{slp} + \tau_i - \gamma_k / T_k^{slp} + \tau_k)^2 + 4\gamma_i \cdot \lambda_k > 0$ . As a result, roots of  $R_i$  in Eq. (12) always exist. Then, we have

$$\bar{T}_i^{slp} = (\bar{R}_i - \gamma_{k'} / T_{k'}^{slp} - \tau_{k'}) / \lambda_{k'}, \quad (14)$$

where  $k' = \arg \max_k \{ (\phi_i^{j,k} + \sqrt{(\phi_i^{j,k})^2 - 4\omega_i^{j,k}}) / 2 \}$ .

#### B. The DSIC Algorithm Design

To deal with the traffic load variance, at any sensor node  $i$ , SIC is performed in a periodical basis or triggered when traffic dynamics are detected. Before the algorithm execution, sensor node  $i$  collects necessary information from neighbor nodes, which includes the current sleep interval  $T_j^{slp}$  of its receiver  $j$ ,  $\lambda_k$ ,  $\gamma_k$ ,  $\zeta_k$ , and  $\tau_k$  of each sender  $k$ . Such information is used to update  $R_i$  and  $T_i^{slp}$  by locally solving the D-SIC problem from Eqs. (6) to (9), or Eqs. (10) to (14) if the strobed preamble technique is adopted. To reduce the communication cost, above parameters can be obtained from regular information exchanges of some underlying services, like link estimations and CTP beacons. After calculating

updated  $\bar{R}_i$  and  $\bar{T}_i^{slp}$ , sensor node  $i$  first checks whether the new  $\bar{R}_i$  is smaller than the current  $R_i$ . If so,  $i$  adjusts its sleep interval to be  $\bar{T}_i^{slp}$ . In addition,  $R_i$  will be replaced by  $\bar{R}_i$  for the next updating. Otherwise,  $i$  takes no action. The detailed description of the DSIC algorithm is given in Algorithm 1. When  $\bar{R}_i < R_i$ , the adjustment of sleep interval will decrease the maximum energy consumption rate in the local region of node  $i$ . Intuitively, by doing so for all other local regions, the network-wide energy bottleneck is eliminated gradually. As a result, the original optimization goal tends to be approached in an iterative manner. A rigorous interpretation to the correctness of our algorithm is given in the following theorem:

*Theorem 4: By the execution of Algorithm 1 at each sensor node, the maximum energy consumption rate in the network approaches to be minimized.*

*Proof:* Line 4 in Algorithm 1 indicates that whenever the sleep interval is updated,  $R_i$  becomes smaller, which results in the decrease of the maximum energy consumption rate decreasing in each local region. On the other hand, the original SIC problem and the D-SIC problem are both convex, and each D-SIC is a sub-problem of SIC. To finish the proof, we assume that the maximum energy consumption rate in the network converges to  $\mathcal{R}$ , which is different from the optimal result  $\mathcal{R}^*$ . Clearly,  $\mathcal{R} > \mathcal{R}^*$ . Now, we prove this theorem by contradiction. If the maximum energy consumption rate converges to  $\mathcal{R}$  by our algorithm, it indicates that there does not exist any  $R_i$  to further reduce the current maximum rate of energy drain in the network, implying  $\mathcal{R}$  be a local minimum point. However, the original problem is convex, such a conclusion yields that  $\mathcal{R}$  must be a global optimal point as well [27], which is a contradiction. ■

### C. Discussions

1) *Multi-receiver scenario:* So far, we have focused on the case, in which each sensor node  $i$  has only one receiver. Such a case corresponds to the packet transmission following a tree-based routing structure. As aforementioned, packets, however, can be transmitted following a DAG as well, in which there may exist more than one potential receiver. Without loss of generality, we assume sensor node  $i$  has  $n_i$  potential receivers.

We can slightly alter our previous analysis and reach a General Distributed SIC (GDSIC) algorithm, which can support multiple receivers in general. GDSIC can be simply extended from the DSIC algorithm, and the basic principle is as follows. Since preambles sent from sensor node  $i$  must cover the sleep interval of each potential receiver  $r_j$  for  $1 \leq j \leq n_i$ , the length of  $i$ 's preamble can be determined by  $\max_j \{T_{r_j}^{slp}\}$ . Thus, the multi-receiver case is accordingly transformed to an equivalent single receiver case as shown by Fig. 5, in which the sleep interval of the single virtual receiver equals to  $\max_j \{T_{r_j}^{slp}\}$ . We can then modify  $T_i^{slp}$  as  $\max_j \{T_{r_j}^{slp}\}$  in line 1 of Algorithm 1 and apply the DSIC algorithm for the sleep interval control. Due to the page limit, the detailed algorithm is omitted and can be found in our technical report [23].

---

### Algorithm 1: The DSIC Algorithm at Sensor Node $i$

---

**Input** : Current  $R_i$  and sleep interval  $T_i^{slp}$ .

**Output:** Updated  $R_i$  and  $T_i^{slp}$ , denoted as  $\bar{R}_i$  and  $\bar{T}_i^{slp}$ .

- 1 Collect  $T_j^{slp}$ , where  $j$  is the receiver of sensor node  $i$ .
  - 2 Collect  $\lambda_k, \gamma_k, \zeta_k$ , and  $\tau_k$  from each sender  $k$ .
  - 3 Locally solve the D-SIC problem from Eqs. (6) to (9) and obtain updated  $\bar{R}_i$  and  $\bar{T}_i^{slp}$ .
  - 4 **if**  $\bar{R}_i < R_i$  **then**
    - 5     Set  $R_i$  to be  $\bar{R}_i$ ;
    - 6     Update the sleep interval  $T_i^{slp}$  by  $\bar{T}_i^{slp}$ ;
    - 7     Inform the updated  $T_i^{slp}$  to its senders;
  - else**
  - 8     Keep both  $R_i$  and  $T_i^{slp}$  unchanged;
  - end**
- 

2) *Extension to receiver-initiated protocols:* As a most representative technique, Low Power Probing (LPP) has been employed in many receiver-initiated protocols. The energy efficiencies of LPL and LPP are mainly different at following two aspects. First, the energy consumption to receive the preamble at the receiver side can be omitted in LPP. Second, the energy consumed for overhearing in LPL should be replaced by obtaining the receiver's predicted wake-up schedule in LPP. After rephrasing the energy consumption rate for each sensor node with LPP based on above two differences, our previous analysis and solution can be seamlessly extended to the receiver-initiated protocols.

3) *SIC for leaf nodes:* Since leaf nodes in the network have no packet senders, they may fail to obtain an effective sleep interval adjustment based on the GDSIC algorithm. To deal with such a marginal case, in our implementation, those sensor nodes adjust their sleep intervals such that their energy consumption rates are approximately equal to their receivers.

## V. EXPERIMENTAL EVALUATION

In previous sections, we elaborate the design principles and important properties of the proposed GDSIC algorithm. In this section, we validate its feasibility and applicability in practice.

### A. Experiment Setting

We implement GDSIC on TelosB motes and use a 50-node testbed to examine its performance. 50 nodes are organized as a  $10 \times 5$  grid<sup>3</sup>. Due to the experimental space limitation, the power of each TelosB mote is set to be the minimum level and the communication range is about 10 centimeters. Starting from the left-top corner, sensors are placed following the bottom-to-top and left-to-right order based on their IDs. GDSIC is implemented at the application layer, which utilizes two major standard components, LPL and CTP, adopted in

<sup>3</sup>Due to the hardware failure, node number 11, in Fig. 6, is excluded and only 49 sensor nodes are finally used in the experiment.

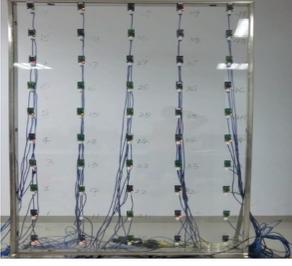


Fig. 6. 10 × 5 grid testbed

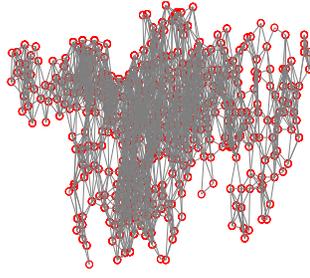


Fig. 7. GreenOrbs Topology

current TinyOS 2.1 package. On the MAC layer, the default protocol, X-MAC, is adopted in the experiment. In the initial five minutes, sensor nodes beacon neighboring nodes to form a stable routing tree rooted at sensor node 0. To increase the depth of the formed routing structure, we manually enforce that the receiver of a sensor node is selected from its adjacent neighbors on the testbed. For example, the parent of node 15 is chosen from nodes 16, 14, 25, 5, 24, 6, 26 and 4. After the initialization phase, sensor nodes inject packets to the network and cooperatively deliver packets to the sink (root) node. The average traffic generation rate is one packet every four seconds and the GDSIC algorithm is triggered every 60 seconds.

## B. Experimental Results

1) *Energy consumption rate vs. duration time*: The experiment lasts 40 minutes on the testbed. Based on the collected data, we observe that after GDSIC executes for 20 minutes, the system performance becomes relatively stable. For a clear presentation, we mainly demonstrate the transition state of the network **after** the initial phase. In Fig. 8, we illustrate energy consumption rates of five representative sensor nodes with hop counts 1, 3, 5, 7, and 9 respectively. Each selected sensor node in Fig. 8 experiences approximately the fastest energy draining speed compared with other peering nodes with the same hop count. Fig. 8 shows that after 800 seconds, energy consumption rates of those sensor nodes converge to around 3.6 mJ/s, and there is no obvious performance variance afterward. At time 1000 seconds, we take a snapshot of the network and conduct an offline computation. The optimal min-max energy fairness is obtained to be 3.2 mJ/s. The important insights obtained from Fig. 8 are two-folds: first, energy consumption rates of sensor nodes in different network positions are well balanced after the network becomes stable, which effectively eliminates the hotspots of energy consumption within the network. Second, GDSIC has a good convergency speed. In particular, after the initial five minutes, the overall energy consumption rates are decreased to be fairly low within first 500 seconds. After several extra iterations, the performance converges eventually. According to Fig. 8, we find that the stabilized energy consumption rates of sensor nodes near the sink node are still slightly greater than other distant sensor nodes in GDSIC and such a performance gap is difficult to be further closed but remains small. Compared

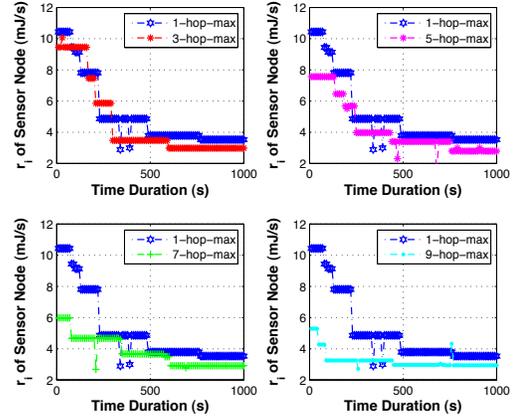


Fig. 8. Energy consumption rate vs. Duration

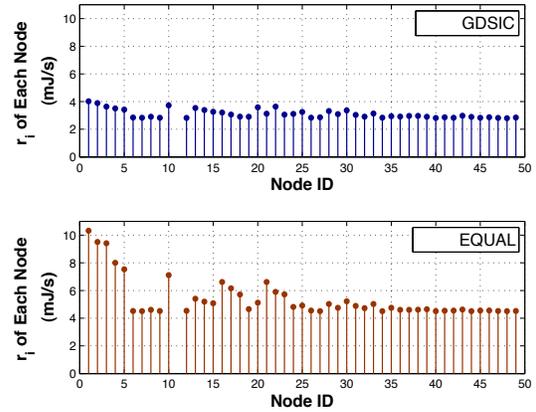


Fig. 9. Snapshot of energy consumption rates

with the equal sleep interval setting policy, the min-max energy fairness has been notably improved by GDSIC.

2) *Snapshot of energy consumption rates*: In Fig. 9, we illustrate the snapshot of the energy consumption rate of each sensor node in GDSIC and compare them with the traditional identical sleep interval setting strategy (EQUAL). According to Fig. 9, we can observe that most sensor nodes in GDSIC achieve similar energy drains after executing the GDSIC algorithm, and only a small number of sensor nodes close to the sink (with heavier traffics) suffer slightly higher energy consumption rates. However, compared with EQUAL, the min-max energy fairness has been improved by GDSIC up to 64.1%, and the obtained energy fairness is close to the optimal result. In addition, the average energy consumption rate in GDSIC also outperforms EQUAL by 37.2%.

## VI. TRACE-DRIVEN SIMULATION EVALUATION

We conduct comprehensive and large-scale simulations to further examine the efficiency and scalability of GDSIC. We evaluate the system performance of GDSIC in comparison with the optimal policy (OPT), the greedy strategy (GREEDY), and the equal sleep interval strategy (EQUAL). In GREEDY,

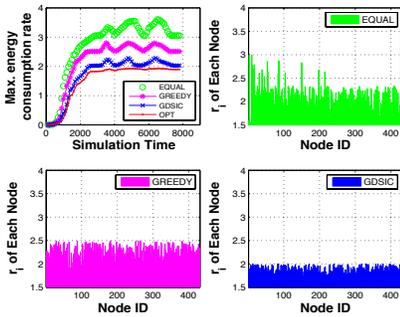


Fig. 10. Maximum energy consumption rates

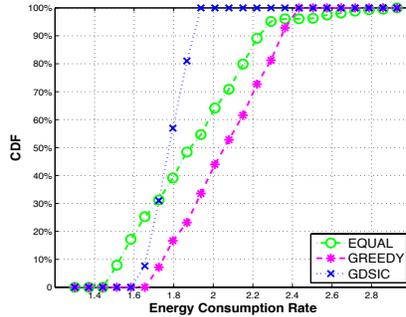


Fig. 11. CDF of energy consumption rates

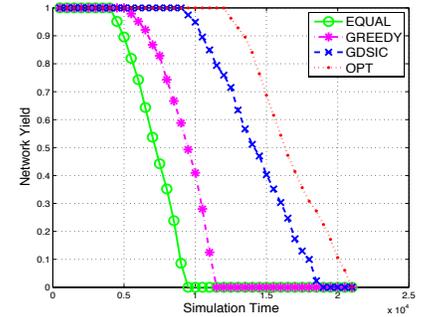


Fig. 12. Network yield

sensor nodes adjust sleep intervals such that their energy consumption rates are set as the average value of their neighbors. To test a realistic network setting, simulations are conducted with a real trace harvested from GreenOrbs [22]. GreenOrbs is a long-term and large-scale wireless sensor network deployed in the forest, which contains 433 nodes and has continuously worked over one year. From the harvested trace over 6 months, we observe that the dynamics of wireless links result in fluctuating of the network topology. To mimic the link estimation for real data transmissions, we filter out lossy links with small RSSI values. In particular, links with the packet reception ratio lower than 30% or RSSI smaller than  $-80dB$  are excluded by the filter. By doing so, we obtain a stable network topology for simulations in Fig. 7. The topology includes 6567 links with relatively good qualities.

#### A. Experimental Setting

In the trace, sensor nodes are deployed in a  $700m \times 200m$  rectangle field with the default transmission power. Parameters of sensor nodes are set based on the Telosb mote specification [28]. Packet retransmissions due to the link loss are considered in the simulation. The sink node is placed at  $(-200.2, 115.7)$  and the default traffic generation rate is one packet every ten seconds. To mimic the traffic dynamics in real applications, we manually trigger the traffic variance and investigate the impact of the traffic dynamics. The default MAC-layer protocol is X-MAC, while we also study the GDSIC strategy over a variety of other asynchronous protocols, adopting the original LPL, strobed preamble and predictive wake-up techniques.

1) *Maximum energy consumption rates:* In Fig. 10, we first investigate the achieved maximum energy consumption rates (min-max energy fairness) with different approaches. We simulate an 8000-second data collection process. During three time intervals of [3500,4000], [4500,5000], and [6000,6500], we double traffic generate rates of sensor nodes in four random regions and each region roughly contains 10% of total sensor nodes. In Fig. 10 (up-left), we observe that EQUAL incurs a much larger min-max energy fairness than both GREEDY and GDSIC all the time. When traffic varies, its min-max fairness fluctuates significantly. As unveiled by Theorem 2, such a fluctuation is mainly caused by the traffic dynamics

since sleep intervals are set to be identical. Compared with EQUAL, GREEDY improves the achieved energy fairness by 16.7% on average and the fluctuation of GREEDY is smoother. However, we can find that there is still a clear gap between GREEDY and OPT, which needs to be closed. In Fig. 10, GDSIC effectively closes such a gap and outperforms EQUAL and GREEDY by up to 32% and 21%, respectively.

To further investigate energy consumption rates of sensor nodes in different systems, we take a snapshot at time 4000 and show the instant energy consumption rate of each sensor node in Fig. 10 (up-left). Similar to Section V, in EQUAL, the energy drains of certain sensor nodes are much faster than other sensors, which will potentially limit the lifetime of the network. The difference between the maximum and minimum energy consumption rates in EQUAL is up to 48.9%. Different from EQUAL, both GREEDY and GDSIC result in a well balanced energy consumption rate over the entire network. Statistics show that differences between the maximum and minimum rates to consume energy in GREEDY and GDSIC are only 29.5% and 14.9%, respectively. However, we notice that the average rate of energy drain in GREEDY is still high, and there is a non-ignorable gap between GREEDY and GDSIC. Fig. 10 provides a good indication that GDSIC has achieved the best performance in terms of both the min-max energy fairness and average energy consumption among three approaches.

2) *CDF of energy consumption rates:* According to Fig. 10, we further illustrate the CDF of energy consumption rates with different strategies in Fig. 11. As expected, energy drains of GDSIC distribute within a narrow interval. In addition, the average value of GDSIC is also the smallest one compared with other two strategies. On contrary, energy consumption rates in EQUAL spread over a wide region from 1.42 to 2.99 mJ/s and a small portion of sensor nodes suffer relatively high speeds of energy consumption, which will limit the lifetime of the network. Another important information delivered from Fig. 11 is that although the distribution of GREEDY is within a narrow region as well, its average value is greater than EQUAL. As the initial rate differences in the network might be large, when the sensor nodes with high energy consumption rates greedily adjust their own sleep intervals,

they unavoidably increase the energy consumption rates of their children. Such a greedy strategy could lead to a sub-optimal result that is far away from the optimal result.

3) *Network yield*: Network yield in Fig. 12 is defined as the percentage of sensor nodes which are reachable from sink node. Fig. 12 shows that even the average energy consumption rate of GREEDY is greater than EQUAL, GREEDY still has a larger network yield than EQUAL all the time, since there is no obvious energy bottleneck in GREEDY. Fig. 12 provides a good indication to the importance of minimizing the maximum energy consumption rate in the network. Different from EQUAL and GREEDY, GDSIC performs closer to OPT all the time and achieves an excellent min-max energy fairness. From statistics, network yield of GDSIC, on average, is greater than EQUAL and GREEDY by 50.8% and 35.9%, respectively.

## VII. CONCLUSION

In this paper, we investigate the problem of achieving the min-max energy fairness in asynchronous duty-cycling sensor networks. We aim at optimal sleep interval control for sensor nodes so as to achieve min-max energy fairness. We propose a mathematical model to describe energy efficiency of such networks and observe that traditional sleep interval setting and control strategies hardly perform well in practice. Towards developing an efficient control strategy, we formulate the SIC problem as a convex optimization problem. By utilizing the convex property, we decompose the original problem, which yields to a distributed algorithm GDSIC. In GDSIC, the network-wide min-max energy fairness can be achieved in a distributed fashion. The proposed solution serves as a unified framework applicable to a variety of underlying asynchronous protocols in wireless sensor networks.

## ACKNOWLEDGEMENT

Zhenjiang Li and Mo Li are supported by the Singapore National Research Foundation under its Environmental & Water Technologies Strategic Research Programme and administered by the Environment & Water Industry Programme Office (EWI) of the PUB, under project 1002-IRIS-09. This work is also supported by the NSFC Distinguished Young Scholars Program under grant No. 61125202, the National Basic Research 973 Program of China under Grant No. 2011CB302705, NSFC Grant No. 60970123, NTU SUG M4080103.020, and NAP M4080738.020.

## REFERENCES

- [1] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in *Proceedings of IEEE Infocom*, 2002, pp. 1567–1576.
- [2] T. Dam and K. Langendoen, "An adaptive energy-efficient mac protocol for wireless sensor networks," in *Proceedings of ACM Sensys*, 2003, pp. 171–180.
- [3] W. Ye, F. Silva, and J. Heidemann, "Ultra-low duty cycle mac with scheduled channel polling," in *Proceedings of ACM Sensys*, 2006, pp. 321–334.

- [4] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of ACM Sensys*, 2004, pp. 95–107.
- [5] M. Buettner, G. Yee, E. Anderson, and R. Han, "X-mac: A short preamble mac protocol for duty-cycled wireless sensor networks," in *Proceedings of ACM Sensys*, 2006, pp. 307–320.
- [6] S. Liu, K.-W. Fan, and P. Sinha, "Cmac: An energy efficient mac layer protocol using convergent packet forwarding for wireless sensor networks," in *ACM Transaction on Sensor Networks (TOSN)*, vol. 5, no. 4, 2009, p. 29.
- [7] Y. Liu, Y. He, M. Li, J. Wang, K. Liu, L. Mo, W. Dong, Z. Yang, M. Xi, J. Zhao *et al.*, "Does wireless sensor network scale? a measurement study on greenorbs," in *Proceedings of IEEE Infocom*, 2011, pp. 873–881.
- [8] Y. Zhu and L. Ni, "Probabilistic approach to provisioning guaranteed qos for distributed event detection," in *Proceedings of IEEE Infocom*, 2008, pp. 592–600.
- [9] S. Rangwala, R. Gummadi, R. Govindan, and K. Psounis, "Interference-aware fairerate control in wireless sensor networks," in *Proceedings of ACM SIGCOMM*, 2006, pp. 63–74.
- [10] Y. Gu and T. He, "Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links," in *Proceedings of ACM Sensys*, 2007, pp. 321–334.
- [11] S. Tang, X. Li, X. Shen, J. Zhang, G. Dai, and S. Das, "Cool: On coverage with solar-powered sensors," in *Proceedings of IEEE ICDCS*, 2011, pp. 488–496.
- [12] Q. Ma, K. Liu, X. Miao, and Y. Liu, "Opportunistic concurrency: A mac protocol for wireless sensor networks," in *Proceedings of IEEE DCOSS*, 2011, pp. 1–8.
- [13] X. Wang, X. Wang, G. Xing, and Y. Yao, "Dynamic duty cycle control for end-to-end delay guarantees in wireless sensor networks," in *Proceedings of IEEE IWQoS*, 2010, pp. 1–9.
- [14] C. Merlin and W. Heinzelman, "Schedule adaptation of low-power-listening protocols for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 5, pp. 672–685, 2009.
- [15] P. Park, C. Fischione, and K. Johansson, "Adaptive ieee 802.15. 4 protocol for energy efficient, reliable and timely communications," in *Proceedings of ACM/IEEE IPSN*, 2010, pp. 327–338.
- [16] G. Challen, J. Waterman, and M. Welsh, "IDEA: Integrated distributed energy awareness for wireless sensor networks," in *Proceedings of ACM Mobisys*, 2010, pp. 35–48.
- [17] A. El-Hoiydi and J.-D. Decotignie, "Low power downlink mac protocols for infrastructure wireless sensor networks," *ACM Mobile Networks and Applications*, vol. 10, no. 5, pp. 675–690, 2005.
- [18] L. Tang, Y. Sun, O. Gurewitz, and D. Johnson, "Pw-mac: An energy-efficient predictive-wakeup mac protocol for wireless sensor networks," in *Proceedings of IEEE Infocom*, 2011, pp. 1305–1313.
- [19] Y. Sun, O. Gurewitz, and D. B. Johnson, "Ri-mac: A receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks," in *Proceedings of ACM Sensys*, 2008, pp. 1–14.
- [20] P. Dutta, S. Dawson-Haggerty, Y. Chen, C. Liang, and A. Terzis, "Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless," in *Proceedings of ACM SenSys*, 2010, pp. 1–14.
- [21] P. Dutta, J. Taneja, J. Jeong, X. Jiang, and D. Culler, "A building block approach to sensor networks," in *Proceedings of ACM Sensys*, 2008, pp. 267–280.
- [22] "GreenOrbs," <http://greenorbs.org>.
- [23] Z. Li, M. Li, and Y. Liu, "Towards energy-fairness in asynchronous duty-cycling sensor networks," *Technical report*, 2012.
- [24] X. Wang, L. Fu, X. Tian, Y. Bei, Q. Peng, X. Gan, H. Yu, and J. Liu, "Converge-cast: On the capacity and delay tradeoffs," *IEEE Transactions on Mobile Computing*, no. 99, pp. 1–1, 2011.
- [25] Y. Liu, Y. Zhu, and L. M. Ni, "A reliability-oriented transmission service in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, pp. 2100–2107, 2011.
- [26] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, "Collection tree protocol," in *Proceedings of ACM SenSys*, 2009, pp. 1–14.
- [27] S. Boyd, *Convex optimization*. Cambridge Univ. Pr., 2004.
- [28] "TelosB," <http://www2.ece.ohio-state.edu/~bibyk/ee582/telosMote.pdf>.