DMS-Based Energy Optimizations for Clustered WSNs

MARYAM BANDARI, ROBERT SIMON, and HAKAN AYDIN, George Mason University

In this article, we consider clustered wireless sensor networks where the nodes harvest energy from the environment. We target performance-sensitive applications that have to collectively send their information to a cluster head by a predefined deadline. The nodes are equipped with Dynamic Modulation Scaling (DMS)-capable wireless radios. DMS provides a tuning knob, allowing us to trade off communication latency with energy consumption. We consider two optimization objectives, maximizing total energy reserves and maximizing the minimum energy level across all nodes. For both objectives, we show that optimal solutions can be obtained by solving Mixed Integer Linear Programming problems. We also develop several fast heuristics that are shown to provide approximate solutions experimentally.

CCS Concepts: • **Networks** → **Network performance analysis**; • **Computer systems organization** → **Sensors and actuators;**

Additional Key Words and Phrases: Dynamic modulation scaling, wireless sensor networks, energy management

ACM Reference Format:
Maryam Bandari, Robert Simon, and Hakan Aydin. 2017. DMS based energy optimizations for clustered WSNs. ACM Trans. Embed. Comput. Syst. 16, 3, Article 86 (April 2017), 28 pages.
DOI: http://dx.doi.org/10.1145/2998179

1. INTRODUCTION

Power management in wireless sensor networks (WSNs) is a critical challenge, given the fact that many of these systems are expected to last for long periods of time, and replacing batteries after they are depleted is, in many cases, expensive or difficult to do [Rault et al. 2014]. To address this problem, there is significant interest in using *energy harvesting* techniques in WSNs. Using environmental energy harvesting technology to charge the storage units that power the wireless sensor nodes offers multiple benefits [Sudevalayam and Kulkarni 2011a]. Energy harvesting reduces or eliminates the need for a direct connection to the electrical distribution infrastructure or the requirement to replace the batteries. It is also a sustainable and environmentally friendly approach to energy production. For nodes installed in harsh and inaccessible environments, energy harvesting provides long-term system life that reduces the need for maintenance.

Deploying energy harvesting systems requires careful selection of a harvesting source, converter and consumption circuits, and the type of energy storage unit [Sudevalayam and Kulkarni 2011b]. Heat, vibration, and solar radiation are among the commonly used harvesting sources. The choice of storage unit depends on the required

This work is supported by US National Science Foundation awards CNS-1421855 and CNS-1116122. The experiments were run on ARGO, a research computing cluster provided by the Office of Research Computing at George Mason University, VA (URL: http://orc.gmu.edu).
Authors' addresses: M. Bandari (Current Address), Google Inc., 1600 Amphitheatre Parkway, Mountain View, CA; email: bandari.m@gmail.com; R. Simon and H. Aydin, Computer Science Department, George Mason University; emails: simon@gmu.edu, aydin@cs.gmu.edu.

DOI: http://dx.doi.org/10.1145/2998179

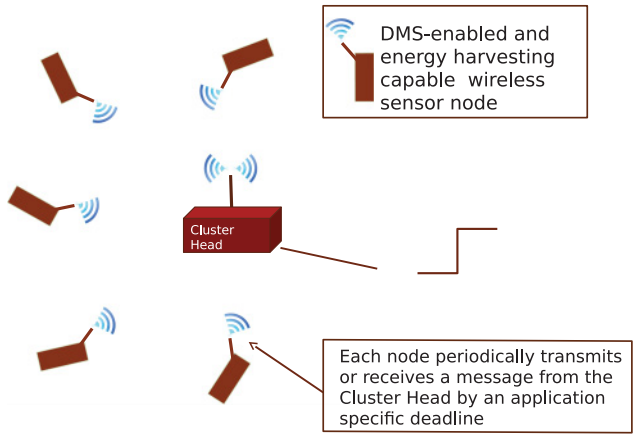


Fig. 1. Wireless energy harvesting nodes equipped with DMS radios communicating with a cluster head.

output voltage, energy density, charge-discharge efficiency, memory effect, and weight. Typically, energy harvesting solutions need to be integrated within the global energy management frameworks implemented in actual systems. Approaches to this problem include cross-layer duty cycling strategies [Glatz et al. 2011] and relay node selection for data transfer [Medepally and Mehta 2010]. It is nevertheless still an open question as to whether energy harvesting methods based on predictable but *noncontrollable* environmental energy sources can be used to support low-power wireless systems that demand performance-sensitive network performance. Due to the variability in the availability of the harvested energy, energy depletion and potential system shutdown might pose serious problems for safety-critical and industrial systems.

This article considers deadline-driven cluster-based wireless sensor networks whose energy storage devices are powered by harvesting noncontrollable but predictable energy sources (such as solar or wind energy). Cluster-based WSNs offer multiple advantages such as increasing scalability, hierarchical routing, and energy saving through data aggregation and minimizing topology maintenance [Mamalis et al. 2009]. As shown in Figure 1, a cluster-based system has a coordinator node that directly communicates with all of the other nodes under its control [Park et al. 2011]. Each sensor node transmits its data to the cluster head periodically. The cluster head has to finish the data collection from all the nodes within a specified deadline. Once it has collected each node's data, the cluster head in turn communicates with the base station directly or through the network to complete data delivery.

In this article, we concentrate on intracluster communication. As a motivating example, consider a target tracking application in which all nodes in a site periodically send their recording to the cluster head. For timely tracking of the target, data delivery in every period must be guaranteed. Delayed data will be overridden by new contents and lose their importance. Another example is utility companies and municipalities that require the constant monitoring of water flow for the purposes of correctly billing customers, as well as for quickly detecting water pipe leakages or breaks. In this example, a WSN could harvest energy from sources such as solar or miniature water dynamos and provide a continuous pipe monitoring service. The objective is to ensure enough time and energy for all nodes along a time horizon to be able to complete their data transmission in a timely manner. This requires an energy-aware time slot assignment for each epoch in the time horizon given the energy profile of the nodes.

Our focus in this article is the use of *Dynamic Modulation Scaling* (DMS) as an energy management technique to support real-time energy harvesting WSNs [Schurgers et al. 2003]. Similar to Dynamic Voltage Scaling (DVS) techniques that use the processor voltage and frequency as a control knob [Pillai and Shin 2001], DMS adjusts the radio modulation levels and constellation sizes, and hence trades off energy expenditure with communication latency [Schurgers et al. 2003]. In our target application, the nodes in the cluster communicate with the coordinator using DMS to save energy. By assigning time slots and modulation levels to individual nodes, the system's energy and timing constraints can be simultaneously satisfied. Signal modulation entails encoding multiple bits on one carrier symbol by varying some properties of the carrier. Changing the modulation level means that the number of bits represented by each symbol, and hence the number of symbols, will change. Increasing (decreasing) the modulation level reduces (increases) the number of symbols that need to be transmitted. An increased modulation level reduces the amount of time required to transmit, at the expense of increased energy consumption. We are motivated to use DMS for several reasons. Since the radio is the dominant energy consumer within wireless sensor nodes, it is more beneficial to manage this tradeoff compared to other energy management techniques, such as DVS [Sudha Anil Kumar et al. 2007]. Further, DMS-enabled radios are supported by embedded wireless standards such as 802.15.4 [Standard 2012].

In our setting, the problem is to determine the time slots and modulation levels that will be used by individual nodes to meet the time constraints and energy neutrality [Kansal et al. 2007] requirements given the predicted harvested energy. The cluster head periodically transmits a beacon packet that assigns time slots and modulation levels to all of the nodes in the cluster. Each node transmits using its assigned time slot and modulation level. We consider two different optimization objectives subject to the aforementioned constraints, namely, maximizing the summation of the node energy levels and maximizing the minimum energy level across all nodes. The first objective aims at improving the system's overall resilience for unexpected extra loads or imperfect harvesting power predictions. The second objective targets achieving a more balanced energy level distribution across the entire network to avoid cases where some nodes' energy reserves may be depleted in order to increase the overall energy of the entire cluster.

We show that both problems can be reduced to the instances of Mixed Integer Linear Programming (MILP) problem. The MILP formulations are optimal in the sense that they will find a *feasible* solution that avoids energy depletion while meeting the communication deadlines, if one exists. Simultaneously, they will maximize the objective under consideration. We also propose and evaluate several fast heuristic algorithms to reduce the computational complexity at the risk of occasionally failing to find some feasible solutions or generating a suboptimal solution in terms of maximizing the target metric. In our solution, energy-rich nodes are allowed to increase their communication speeds, thereby providing the low-energy nodes with large time slack and ability to slow down without violating the timing constraints. We experimentally evaluate the tradeoffs between the optimal but computationally intensive MILP solutions and the simpler heuristic approaches. Our results indicate that the heuristic algorithms that allow individual nodes to apply different modulation levels offer advantages over those that do not. While much simpler, a heuristic based on the *uniform* modulation level assignment fails to generate feasible solutions for a significant portion of the parameter spectrum.

2. RELATED WORK

Our work focuses on real-time cluster-based WSNs under the control of a central coordinating node. Because of the performance requirements of our targeted application

class, we use TDMA to schedule packet transmissions. There has been a significant body of work studying the use of TDMA assignment methods for WSNs. For instance, the work in Shen et al. [2014] uses a preemptive method to give high-priority traffic preference over low-priority traffic. Since TDMA mechanisms are commonly used in industrial standards such as 802.15.4e, Wireless Hart, and ISA 100.11a, many authors have proposed and analyzed algorithms for slot assignment within these protocols [Hanzálek and Jurčík 2010; Toscano and Bello 2012; Ding and Hong 2013]. The goals of these papers revolve around issues such as reducing delay, improving throughput, or maintaining fairness. None of this earlier work considers the potential benefits of DMS or the implications of energy harvesting.

There has also been some research into the problem of TDMA or TDMA-like WSNs that use energy harvesting. For instance, Iannello et al. [2012] use Markov modeling to predict the probability of packet delivery under a number of energy harvesting conditions. Luo et al. [2013] propose the use of reserving slots for energy harvesting followed by packet transmission. Michelusi and Zorzi [2013] cast the problem as a multiaccess game and describe the Symmetric Nash Equilibrium. Michelusi and Zorzi [2015] essentially extend this approach using utility maximization. This previous work also does not take into consideration the possibility of energy harvesting.

For many WSN applications, the deployment environment impedes manually replacing the nodes' batteries and therefore a sustainable energy source must be considered. This includes both habitat and environmental monitoring [Mainwaring et al. 2002] and industrial [Gungor and Hancke 2009] applications. Potential environmental sources of energy include solar, temperature differences, electromagnetic energy, airflow, and piezoelectric. Gorlatova et al. [2010] review the challenges of solar and piezoelectric sources and suggest a cross-layer solution for energy saving. The authors also report the measurements of indoor light as an energy source for indoor WSN applications. The measurements show a high variability in harvested energy, which emphasizes the need for an energy management technique to save the unused energy of burst times for scarce-energy situations. Relying solely on renewable sources introduces design difficulties including maintaining a minimum service level for the nodes, constant processing and communication power, and reliable data delivery. The work presented in Gubbi et al. [2013] reviews a number of performance constraints of WSNs and the challenges of meeting those criteria when the node only relies on the ambient energy sources. Vigorito et al. [2007] address the challenges of preserving energy neutrality, maximum performance, and minimum service level variation and propose a duty cycling solution to achieve the desirable objective. We consider a more complex cluster setting where energy saving is collaboratively pursued by all of the nodes in the cluster, as opposed to a single node. The power industry has recently embraced WSN as a cheap and more reliable substitute of the conventional monitoring devices. Smart grid management systems are a good example of WSN applications that benefit from Radio Frequency (RF)-based wireless energy transfer as its energy source [Erol-Kantarci and Mouftah 2012].

Despite its potential benefits, there has been relatively little attention paid to DMS for wireless sensor networks. An important and highly cited early work on DMS is Schurgers et al. [2003]. In order to schedule packets, the authors suggest deploying DMS in a data link layer accompanied with TDMA. Packets are scheduled using Earliest Deadline First and are assigned a static scaling factor based on their maximum size. This requires sending a control packet to communicate the factor with the receiver. For a single packet in a time-variant channel, the authors propose a DMS approach based on sampling the channel quality.

Subsequent DMS research has focused on issues such as choice of modulation scheme. Cui et al. [2005] derive detailed circuit and transmission energy equations for both

Multiple Quadrature Amplitude Modulation (M-QAM) and Multiple Frequency Shift Keying (MFSK) over additive white Gaussian noise (AWGN) channels. Lowering the transmission speed may also be desirable in nonorthogonal multiple-access schemes since higher speeds create more interference with other users. Miao et al. [2010] proposed a link adaptation technique for Orthogonal Frequency Division Multiplexing (OFDM) by assigning bit rates to orthogonal channels, which maximizes the amount of data sent with a given amount of energy. A similar study in Li et al. [2007] derives equations for energy consumption for integrated transceiver front ends and analyzes the effects of occupied signal bandwidth, peak-to-average ratio (PAR), symbol rate, constellation size, and pulse-shaping filter roll-off factor. Their experiments on the modulation level confirm the convexity of the energy function, used in this work. DMS is applied in Zhang and Chanson [2005] as the power management scheme in energy harvesting time-constrained wireless sensor networks, with the objective of maximizing data throughput. The model optimizes energy for a single node that transmits to multiple receivers. The authors confirmed the convex relationship between the energy consumption and modulation level in Binary Phase-Shift Keying (BPSK).

Our approach is to focus on the common case of *multiple collaborative sensors* reporting to a single base station. In addition, we provide a more general framework of energy harvesting in which the amount of predicted energy for each node changes over time. DMS is applied at intercluster and intracluster levels of communication using collaborative communication in Zhou et al. [2008]. Their numerical results show the advantage of applying DMS. Their model does not include energy harvesting or battery size limitation.

The work in Zhang et al. [2013] uses both DMS and DVS and targets WSNs arranged in tree topologies. This approach aims to maximize system resilience to network-wide workload bursts. Our current work focuses on cluster networks, topologies more suited to industrial control systems. Further, our work guarantees energy neutrality when possible.

The research presented in Fateh and Govindarasu [2015] addresses precedence, interference, and timing constraints to minimize joint communication and computation energy. The authors propose a heuristic method that converts the problem into a graph model and assigns slots with maximum parallelism in order to achieve the maximum static slack. Fateh and Govindarasu [2013] study dynamic slack reclamation in applications that discard redundant data by simply sending the header. Nodes probe the channel after the transmission time of the header of their predecessors in the schedule. If the channel was found free, they apply DMS to reclaim the dynamic slack due to redundancy.

Compared to Fateh and Govindarasu [2013, 2015], our model avoids interference by scheduling node transmissions using TDMA. Our target application has a star topology where all nodes talk directly to the cluster head and they share a collective deadline. The choice of underlying structure, a.k.a. topology, protocol, and our scheduling algorithm, leads to a collaborative system life improvement. In addition, energy harvesting adds a whole new dimension because the energy neutrality condition must be satisfied. To the best of our knowledge, this is the first work that facilitates boosting the energy reserves of one node by using excessive energy reserves of its neighbors in the network by transmission rate adjustment.

The most current, widely available type of DMS-capable radios are used in wireless military communication systems. These systems generally use *Software Definable Radio (SDR)* stacks to control basic radio properties. The SDR changes the modulation type in software. For example, Thales Line-of-Sight (RFE 4145) and Satellite (Modem 21) systems use a scheme called Adaptive Coding Modulation (ACM), which changes modulation speeds (along with other parameters) in response to changing

conditions [Thales 2015]. This modem uses combinations of modulation schemes and code rates for 12 different levels. Similarly, the smart meter industry started to adapt DMS-capable systems. For instance, Sensus uses the iCon family of products, which supports seven different modulation levels [Sensus 2009].

It is expected that the DMS capability will be available in radios used in low-power WSNs in the near future. In fact, some existing sensor nodes already offer multiple modulation schemes (but not full DMS capability within a given modulation scheme). For instance, CC1100 [2014] uses 2-FSK, MSK, OOK, and ASK, while CC2500 [2014] uses 2-FSK, 2-GFSK, MSK, and OOK schemes.

3. SYSTEM MODEL

In our system model, each sensor node consists of an energy harvesting element such as a solar panel or a wind turbine, an energy storage unit such as a rechargeable battery or super capacitor, a CPU, and a DMS-capable radio. Nodes communicate directly with a gateway or cluster head, which serves as the system coordinator. The gateway will often, but not always, be connected to a back-end network. Compared to the nodes, the gateway has ample computational capabilities and is powered either by easily replaceable batteries or a main power supply. Each node periodically senses its environment and after a processing step sends these readings in the form of data packets to the gateway over wireless channel. We assume the requirement for real-time communication between the gateway and each node, which is specified by a fixed deadline. To ensure real-time performance and to avoid channel contention, the cluster uses frame-based transmission scheduling techniques such as Time Division Multiple Access (TDMA) or Guaranteed Time Slot (GTS) assignments during the contention-free period of the ZigBee/802.15.4 super-frame [Park et al. 2011].

The clustered topology and the presence of the computationally powerful gateway enable a coordinated set of DMS node-level settings that guarantee, when possible, both real-time communication and energy neutrality constraints. The *energy neutrality* condition [Kansal et al. 2007] requires that each node is able to sustain itself by harvesting energy and regulating its energy consumption according to the initial energy reserves as well as predicted harvesting profiles. Time is divided into a set of energy harvesting *epochs*. For concreteness and without loss of generality, we consider harvesting solar power and consider a 24-hour scheduling horizon. An epoch is a time window during which the rate of harvested energy is approximated to be constant. For example, in a period of 24 hours, 48 epochs of length 30 minutes exist. Using techniques such as those discussed in Bartolini et al. [2012], the gateway possesses a function or table that predicts for each epoch the amount of energy that will be harvested by the nodes in the cluster. The gateway uses its knowledge of the currently available stored energy and the predicted amount of the newly harvested energy to assign modulation levels at each node for the entire epoch. The solution must make sure that all deadlines are met and the energy neutrality condition is satisfied, whenever possible.

We assume that the WSN has been properly dimensioned and under typical operating conditions the link is not overloaded (e.g., the utilization never approaches 100%). This is the common assumption in most research that focuses on link layer WSN issues [Iannello et al. 2012; Luo et al. 2013; Michelusi and Zorzi 2013, 2015]. We assume the existence of slack time in the schedule with the maximum modulation level. Our suggested framework enables collaboration between energy-rich and energy-poor nodes to increase the energy savings.

The gateway is responsible for running the algorithms and then informing each node about its modulation level for the duration of the epoch. As shown in Figure 2, each epoch consists of a repeating set of *super-frames*, each of length D. The super-frame itself is divided into a set of variable-length transmission slots assigned to

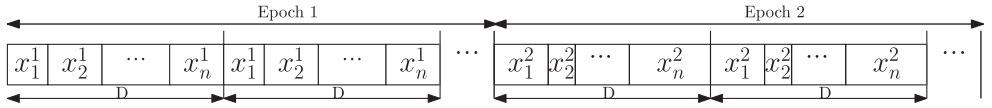


Fig. 2. Transmission slots in super-frames of length D in consecutive epochs.

different nodes. x_i^j represents the transmit time the scheduler assigns to node i in each super-frame during epoch j, which is adjustable through DMS. The transmission time for a node is identical for different super-frames within an epoch but may vary for different epochs. Hence, with γ super-frames within each epoch, node i receives a total of $t_i^j = x_i^j \cdot \gamma$ transmission time during epoch j. Although not shown in the figure, in practice each super-frame could also include space for a gateway-generated beacon used for management purposes.

The energy consumption of a node during an epoch is the sum of sensing, processing, and radio transmission energy figures. Following the common assumption in WSN research (e.g., Restuccia and Das [2016], Varshney et al. [2015], Xie et al. [2013], Zhang et al. [2015], and Li et al. [2013]), we assume sensing and processing energy components are constant and negligible compared to radio (communication) power. Hence, in the rest of this article, we focus on radio power.

Energy consumption due to the radio usage has two components: *electronic circuitry power* and *transmit (receive) power* [Schurgers et al. 2003]. Electronic circuitry power consumption P_e is due to activities such as filtering, modulation, and up-converting. It is linearly proportional to the symbol rate R_s and can be expressed as:

$$P_e = C_e \cdot R_s,$$

where C_e is a radio-specific constant. The transmit power P_s is a function of the number of bits per symbol b, also referred to as the *modulation level* [Schurgers et al. 2003; Fateh and Govindarasu 2015], and is given by

$$P_s = (C_s \cdot \phi(b)) \cdot R_s.$$

For DMS, the modulation level is taken from a set of size B,

$$\{b_1, b_2, \ldots, b_B\}, \text{ where } b_i < b_{i+1}, i = 1, \ldots, B-1.$$

The function $\phi(b)$ depends on the modulation scheme. For example, for QAM, $\phi(b) = 2^b - 1$ [Schurgers et al. 2003]. The coefficient C_s is constant with respect to the modulation level and depends on the receiver implementation, operating temperature, distance, and propagation environment. The time required to send 1 bit is calculated as

$$T_{bit}(b) = \frac{1}{R_s \cdot b}. \tag{1}$$

Assuming QAM and combining the previous expressions for electronic circuitry power, transmit power, and transmit time, the energy to send 1 bit can be derived as [Schurgers et al. 2003]

$$E_{bit}(b) = (P_s + P_e) \cdot T_{bit}(b) = \frac{C_s \cdot (2^b - 1) + C_e}{b}. \tag{2}$$

We observe that the energy consumption is independent of the symbol rate, and decreasing the modulation level reduces energy consumption substantially, at the expense of longer transmission times. This observation demonstrates the potential importance of DMS.

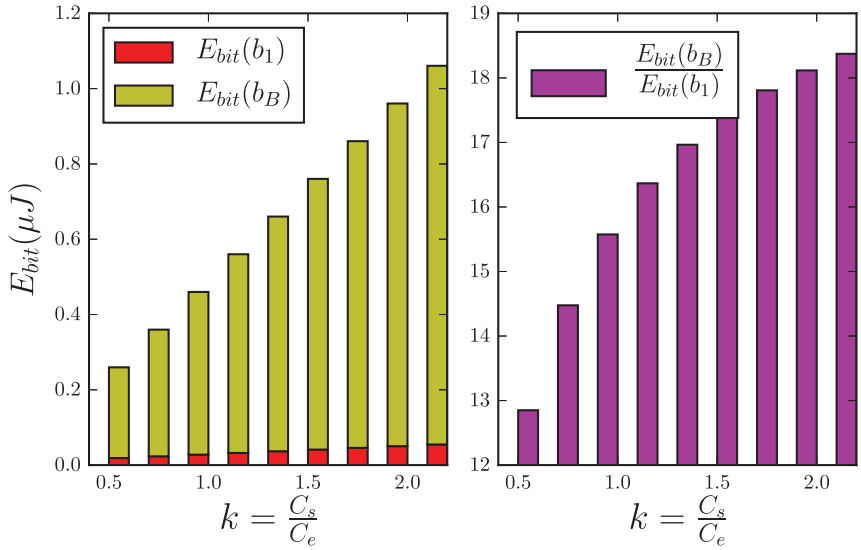


Fig. 3. DMS effectiveness as a function of $k = \frac{C_s}{C_e}$.

It is important to note that the potential of DMS in terms of energy savings is a function of the ratio $\frac{C_s}{C_e}$. Specifically, by defining $k = \frac{C_s}{C_e}$, we can rewrite the expression for the energy needed to transmit a bit as

$$E_{bit}(b) = \frac{k \cdot C_e \cdot (2^b - 1)}{b} + \frac{C_e}{b}.$$

The first component in the previous equation represents the dynamic communication energy, $E_s = P_s \cdot T_{bit} = \frac{k \cdot C_e \cdot (2^b - 1)}{b}$, while the second component represents the energy consumed by the electronic circuitry, $E_e = P_e \cdot T_{bit} = \frac{C_e}{b}$. We illustrate the impact of k by an example. In Figure 3, on the left, the energy per bit $E_{bit}()$ is shown as a function of $k = \frac{C_s}{C_e}$, for the lowest and highest modulation levels b_1 and b_B, when we keep C_e constant. On the right subfigure of Figure 3, we show the ratio of $\frac{E_{bit}(b_B)}{E_{bit}(b_1)}$. As can be seen in the figures, increasing k results in greater energy-savings potential by adjusting the modulation level.

There is an *effective modulation level* below which DMS is no longer beneficial, as shown in Schurgers et al. [2003]. Its value can analytically be determined by setting the first derivative of the radio energy to zero. Similarly, as Cui et al. [2005] indicates, the bit error rate tends to increase with decreasing modulation levels. Using the analytic formulae (as those proposed in Cui et al. [2005]), it is possible to compute the minimum modulation level that is acceptable to support a given, acceptable bit error rate. Consequently, we are assuming that the minimum modulation level b_1 is determined by considering the maximum value among the effective modulation level, the lowest level that still supports the target bit error rate, and the lowest level supported by the modulation scheme.

M packets each of length ρ bits are sent by each node in every super-frame. Each node is assigned a fixed modulation level in a given epoch, but the assignment may vary from epoch to epoch and from node to node. Since there are γ super-frames within each epoch, during the j^{th} epoch, for the i^{th} node with modulation level of b_i^j, the total

communication time t_i^j and energy consumption e_i^j are given by

$$t_i^j = (\gamma \cdot M \cdot \rho) \cdot T_{bit}(b_i^j)$$

and

$$e_i^j = (\gamma \cdot M \cdot \rho) \cdot E_{bit}(b_i^j),$$

respectively. Our algorithms assume that if all the nodes use the maximum modulation levels, in the absence of the energy constraints, it is possible to meet the deadline in each super-frame; that is,

$$n \cdot \frac{M \cdot \rho}{R_s \cdot b_B} \leq D.$$

No solution can exist if that condition is not satisfied.

Due to the short duration of an epoch (e.g., 15 to 30 minutes), we assume that the harvesting power is constant in an epoch [Moser et al. 2008]. There are several techniques for forecasting how much energy will be harvested during the epoch from uncontrollable but predictable environmental sources, including using an autoregressive filter [Kansal et al. 2007], a round-based approach [Kinoshita et al. 2008], or an exponentially weighted moving average method [Kansal and Srivastava 2003]. In order to consider environmental temporary conditions, Noh et al. [2009] introduce a short-term factor as the ratio of observed and predicted energy in the last epoch. This, combined with a degrading weight factor, is applied to the predicted model to improve the accuracy. Our approach is independent of the prediction technique but assumes its availability.

P_i^j denotes the harvested power prediction during epoch j, for node i. The process of energy conversion from the energy harvesting panel to electrical energy consumable by the node components is subject to several harvesting, conversion, storing, and consumption inefficiencies. We assume that these factors are already analyzed and factored in the reported value of P_i^j. Therefore, the harvested energy of node i in epoch j is obtained as $c \cdot P_i^j$, where c is the duration of a single epoch. On node i, the harvested energy is stored in the energy storage unit with capacity J_i. The energy level at the beginning of the first epoch and the target energy at the end of the operation period (with Υ epochs) are denoted by E_i^0 and E_i^{Target}, respectively. The energy level of node i at the end of epoch j is denoted by L_i^j.

4. OPTIMIZATION PROBLEMS

We now formulate the problem we are addressing. Assume there are Υ consecutive epochs for which we have available energy predictions. Given a set of n wireless energy harvesting nodes, each equipped with DMS-capable radios and operating in an environment with a known energy harvesting profile, we aim to determine the modulation level for each node in every epoch, so as to satisfy our objective, while *guaranteeing the deadline in every communication super-frame* and *ensuring the energy neutrality throughout the operation*. We considered two objective functions for our clustered sensor network settings as discussed in the following subsections.

4.1. Objective 1: Maximizing Total Remaining Energy

The first objective is to *maximize the sum of energy levels at the end of the* Υ *epochs*. Specifically, the optimization problem can be formulated as

$$\text{Maximize} \qquad \sum_{i=1}^{n} L_i^{\Upsilon} \tag{3}$$

$$\text{Subject to} \quad \sum_{i=1}^{n} x_i^j \leq D, \qquad \forall j : 1 \leq j \leq \Upsilon \tag{4}$$

$$0 < L_i^j \leq J_i, \qquad \forall i, j : 1 \leq i \leq n,\ 1 \leq j \leq \Upsilon - 1 \tag{5}$$

$$E_i^{Target} \leq L_i^{\Upsilon} \leq J_i \qquad \forall i : 1 \leq i \leq n. \tag{6}$$

The objective is to maximize the sum of remaining energy levels across all nodes at the end of the Υ epoch, $\sum_{i=1}^{n} L_i^{\Upsilon}$, as specified in Equation (3). The real-time communication performance requirement is encoded in the constraint set in Equation (4): all nodes should complete their transmission within time D, the length of the super-frame. The choice of the objective function is to increase the system's resilience against temporary changes in energy resources and potential prediction inaccuracies.

In addition, it is necessary to avoid both battery overflow and underflow conditions, as well as guarantee energy neutrality (constraint sets in Equations (5) and (6)). Prevention of the battery underflow condition ensures the battery level never drops to zero. By preventing the battery overflow, we make sure that the battery level at the end of each super-frame does not exceed the node's storage capacity. Finally, the energy neutrality constraint will ensure that the energy level at the end of the last epoch should not be less than a given target level E_i^{Target}. For a single node, a natural choice is $E_i^{Target} \geq E_i^0$, which makes sure that the system is able to sustain itself by relying on the harvested power only [Kansal et al. 2007].

Given that the modulation level for a given node i is only one of the B distinct values, we can use a binary indicator variable α_{il}^j to specify whether the l^{th} modulation level was selected for node i in epoch j or not. In fact, after a series of additional algebraic manipulations, it is possible to re-encode the problem as an instance of a *Mixed Integer Linear Programming Problem*. The solution to this problem is denoted as *Opt1* in the rest of this article. The full steps of the derivation can be found in the appendix.

4.1.1. Heuristic Algorithms. The Mixed Integer Linear Programming is known to be NP-hard; however, problem instances of moderate size can be solved by existing optimization packages. Nonetheless, it is very desirable to develop schemes that run fast and yield good performance while satisfying all the constraints. For this purpose, we developed and evaluated two fast algorithms, which are described next.

Uniform Modulation Level Assignment

A basic heuristic is to assign equal modulation levels (transmission times) for all nodes and epochs. This *uniform* assignment is not fully energy aware since it does not discriminate between energy-poor and energy-rich nodes. Consequently, the slot assignment is the same for all epochs. Because the energy consumption is an increasing function of the modulation level, the problem reduces to finding the smallest modulation level that meets the deadline in a super-frame:

$$b^* = \min \left\{ b_k | n \cdot \frac{M \cdot \rho}{R_s \cdot b_k} \leq D \right\}.$$

The term $n \cdot \frac{M \cdot \rho}{R_s \cdot b_k}$ in the previous equation is the total time that it takes for all nodes in the cluster to complete their transmission with the uniform modulation level b_k. We are interested in the smallest value of b_k, called b^*, that meets the deadline constraint since further increasing the modulation level only increases the energy consumption, which is clearly against the nature of the objective function. The minimum time-feasible *uniform* modulation level is also energy feasible if and only if it also meets the energy constraints. Otherwise, no larger value of modulation level will meet those constraints.

ALGORITHM 1: *Uniform* Modulation Level Assignment

1: $r = 1$
2: **while** $(n \cdot M \cdot \rho) \cdot T_{bit}(b_r) > D$ **do**
3: $r = r + 1$
4: $e(b_r) = (\gamma \cdot M \cdot \rho) \cdot E_{bit}(b_r)$
5: **for** $i = 1 : n$ **do**
6: **for** $j = 1 : \Upsilon$ **do**
7: $L_i^j = \min\{J_i, L_i^{j-1} + c \cdot P_i^j - e(b_r)\}$
8: **for** $i = 1 : n$ **do**
9: **for** $j = 1 : \Upsilon - 1$ **do**
10: **if** $L_i^j < 0$ **then**
11: **return** (r, Infeasible)
12: **for** $i = 1 : n$ **do**
13: **if** $L_i^\Upsilon < E_i^{Target}$ **then**
14: **return** (r, Infeasible)
15: **return** (r, Feasible)

Smaller modulation levels ($b < b^*$) will not meet the time constraints. Evaluating b^* may involve scanning through all choices of modulation levels in the worst case. For a candidate b value, the scheme (whose pseudo-code is provided in Algorithm 1) computes the energy levels at the end of all the epochs for all nodes (the L_i^j values) and checks the energy neutrality conditions. Note that in Line 7, the remaining energy for each node (L_i^j) is updated by taking into account the battery capacity (J_i), the remaining energy at the end of the previous epoch (L_i^{j-1}) , the total harvested energy ($c \cdot P_i^j$), and the energy consumed during the epoch (($e(b_r)$). The overall time complexity is linear in the number of nodes and the length of the operation period, specifically, $O(B+n \cdot \Upsilon)$. The maximum number of modulation levels in a modulation scheme (B) is a small constant (e.g., $B = 5$ in QAM), which yields a simplified complexity figure of $O(n \cdot \Upsilon)$. The algorithm returns the index of the common modulation level and an indicator showing if the solution was energy feasible or not.

Greedy Modulation Level Assignment

One major drawback of *Uniform* is that some nodes are prevented from further lowering their modulation levels (i.e., they do not exploit the maximum transmission slack that can be used by DMS). In fact, there may be problem instances where the energy feasibility cannot be satisfied by assigning a uniform modulation level, while reducing the levels of individual nodes may lead to a feasible solution.

Our *Greedy* scheme (Algorithm 2) takes the output of the *Uniform* heuristic as the base case. Specifically, the lowest uniform modulation level b_r that meets the deadline constraint is taken as the initial assignment for all the nodes. Then the algorithm iterates over all the epochs and all the nodes and attempts to reduce the modulation levels of selected nodes by exactly one level (i.e., their modulation levels are set to b_{r-1}). In each iteration, and for every epoch, the algorithm tentatively selects the nodes with the minimum amount of remaining energy for the purpose of slowdown. Decreasing the modulation level of a node within one epoch only by one level gives other nodes a chance to reduce their transmission speed as well. This is further justified by the convexity of the energy consumption function and the objective of saving energy as much as possible. After such adjustments, the target energy constraint and feasibility conditions are rechecked.

Note that the algorithm allows each node to scale down its modulation level by only one level in each epoch, below the level suggested by *Uniform*. This helps to reduce the

ALGORITHM 2: *Greedy* Modulation Level Assignment

1: Set $r = Uniform()$
2: Set $b_i^j = b_r \ \forall i, j : 1 \leq i \leq n, 1 \leq j \leq \Upsilon$
3: **if** $r == 1$ **then**
4: **if** *Uniform* returned Infeasible **then**
5: **return** (Infeasible)
6: **else**
7: **return** $(\{b_i^j\}) \forall i, j : 1 \leq i \leq n, 1 \leq j \leq \Upsilon$
8: /* $r > 1$ and *Greedy* starts iterations */
9: $t(b_r) = \gamma \cdot M \cdot \rho \cdot T_{bit}(b_r)$
10: $t(b_{r-1}) = \gamma \cdot M \cdot \rho \cdot T_{bit}(b_{r-1})$
11: **for** $j = 1 : \Upsilon$ **do**
12: **for** $i = 1 : n$ **do**
13: $L_i^j = \min\{J_i, L_i^{j-1} + c \cdot P_i^j - e_i^j\}$
14: Sort all L_i^j values for all the nodes in the j^{th} epoch
15: $Slack = D \cdot \gamma - \sum_{i=1}^{n} t_i^j$
16: $Q = \{1, \ldots, n\}$
17: **while** $Q \neq \emptyset$ and $Slack \geq (t(b_{r-1}) - t(b_r))$ **do**
18: $Slack = Slack - (t(b_{r-1}) - t(b_r))$
19: $index =$ node index with minimum L_i^j value in Q
20: $b_{index}^j = b_{r-1}$
21: $L_{index}^j = \min\{J_i, L_{index}^{j-1} + c \cdot P_{index}^j - e_{index}^j\}$
22: $Q = Q - \{index\}$
23: **for** $i = 1 : n$ **do**
24: **if** $L_i^j < 0$ **then**
25: **return** (Infeasible)
26: **for** $i = 1 : n$ **do**
27: **if** $L_i^{\Upsilon} < E_i^{Target}$ **then**
28: **return** (Infeasible)
29: **return** $(\{b_i^j\}) \forall i, j : 1 \leq i \leq n, 1 \leq j \leq \Upsilon$

complexity; moreover, it is based on the observation that if the nodes were allowed to scale down their modulation levels further by consuming a larger portion of the slack time, that would be less effective than allocating the slack to other nodes in the same epoch. In other words, *Greedy* attempts to allocate the slack time more evenly among the nodes resulting in more collective energy saving, since the objective is to minimize the total remaining energy.

The *while* loop in line 17 of Algorithm 2 selects one node to reduce transmission speed in every iteration, and the existing slack is reduced accordingly after such a selection. The iterations terminate when either all node modulation levels are already scaled (breaking the condition $Q \neq \emptyset$) by one level or the slack is reduced to a level that does not allow further scaling down (breaking the condition $Slack \geq (t(b_{r-1}) - t(b_r))$). Consequently, the *while* loop terminates after at most n iterations. Since the rest of the algorithm has a predetermined number of operations (due to the *for* loops), the algorithm is guaranteed to terminate.

The algorithm invokes *Uniform* at the beginning. Moreover, it requires sorting the remaining energy levels at the end of each epoch. Each sort takes $O(n \log(n))$ time, and there are Υ total epochs. At most $n - 1$ nodes will require changing modulation levels, the complexity of which is outweighed by the sorting complexity. As a result, the algorithm has an overall time complexity of $O(\Upsilon \cdot n \cdot \log(n) + B)$. For a given modulation scheme, B is a small constant and the complexity can be simplified as $O(\Upsilon \cdot n \cdot \log(n))$.

4.2. Objective 2: Maximizing Minimum Remaining Energy

The second objective we studied is to *maximize the minimum remaining energy across the cluster*, at the end of the operation period, that is, after Υ epochs. This objective may be more suitable than maximizing the total energy for settings where it is important to have a more balanced remaining energy distribution across all the nodes. It will also help to prepare the cluster to be more resilient against harvesting power prediction imprecisions in future epochs. Formally, the problem is formulated as

$$\text{Maximize} \min_{1 \le i \le n} L_i^{\Upsilon}. \tag{7}$$

The performance (Equation (4)), minimum remaining energy (Equation (5)), and target energy (Equation (6)) constraints remain as described earlier. Although the Max-Min objective is not immediately linear in terms of the unknown variables, a transformation helps us to obtain another instance of the mixed integer linear programming problem.

Specifically, we introduce an auxiliary variable σ and change the objective function to maximize σ:

$$\text{Maximize} \quad \sigma. \tag{8}$$

The auxiliary variable represents the minimum of our main variables (L_i^{Υ}) at the end of the final epoch, Υ. We add extra constraints to the formulation to enforce this property:

$$\sigma \le L_i^{\Upsilon} \qquad \forall i : 1 \le i \le n. \tag{9}$$

It is easy to see that by maximizing σ, we are maximizing the minimum of the remaining energy levels. Consequently, the new problem is equivalent to

$$\begin{aligned}
&\text{Maximize} && \sigma && \\
&\text{Subject to} && \sum_{i=1}^{n} x_i^j \le D, && \forall j : 1 \le j \le \Upsilon \\
& && 0 < L_i^j \le J_i, && \forall i, j : 1 \le i \le n,\ 1 \le j \le \Upsilon - 1 \\
& && E_i^{Target} \le L_i^{\Upsilon} \le J_i, && \forall i : 1 \le i \le n \\
& && \sigma \le L_i^{\Upsilon} && \forall i : 1 \le i \le n.
\end{aligned}$$

By introducing the binary indicator variables as done for Objective 1, we can obtain another instance of an MILP problem. The optimal solution to this problem, denoted by *Opt2*, can be obtained by resorting to the optimization tools such as Cplex, Gurobi, Lingo, and AMPL, for moderate-size instances.

4.2.1. Heuristic Algorithms. We looked for fast heuristic algorithms for Objective 2 as well. *Uniform* is equally applicable for the objective of maximizing the minimum remaining energy with its simplicity and low complexity. However, our experiments showed that *Greedy*'s performance is subpar for this objective. We designed a new algorithm, *Aggressive*, to find a fast approximate solution for our second objective. *Aggressive* is designed specifically for maximizing the minimum remaining energy at the end of the last epoch by allowing the energy-poor nodes to apply DMS multiple times as described in the following.

Aggressive Modulation Level Assignment

Greedy, which was proposed for Objective 1, has two characteristics: the maximum modulation level is the one calculated by *Uniform*, and each node can only be chosen once in each epoch for scaling down the modulation level. Our new heuristic *Aggressive*

ALGORITHM 3: *Aggressive* Modulation Level Assignment

1: Set $b_i^j = b_B \; \forall i, j : 1 \leq i \leq n, 1 \leq j \leq \Upsilon$
2: /* *Aggressive* starts iterations */
3: $t(b_B) = \gamma \cdot M \cdot \rho \cdot T_{bit}(b_B)$
4: **for** $j = 1 : \Upsilon$ **do**
5: **for** $i = 1 : n$ **do**
6: $L_i^j = \min\{J_i, L_i^{j-1} + c \cdot P_i^j - e_i^j\}$
7: Sort all L_i^j values for all the nodes in the j^{th} epoch
8: $Slack = D \cdot \gamma - \sum_{i=1}^{n} t_i^j$
9: $Q = \{1, \ldots, n\}$
10: **while** $Q \neq \emptyset$ and $Slack > 0$ **do**
11: $index =$ node index with minimum L_i^j value in Q
12: $b_h =$ modulation level of the node with minimum L_i^j value in Q
13: **if** $Slack + t(b_h) - t(b_{h-1}) \geq 0$ **then**
14: $Slack = Slack + t(b_h) - t(b_{h-1})$
15: $b_{index}^j = b_{h-1}$
16: $L_{index}^j = \min\{J_i, L_{index}^{j-1} + c \cdot P_{index}^j - e_{index}^j\}$
17: **if** $b_{index}^j == b_1$ **then**
18: $Q = Q - \{index\}$
19: **else**
20: $Q = Q - \{index\}$
21: **for** $i = 1 : n$ **do**
22: **if** $L_i^j < 0$ **then**
23: **return** (Infeasible)
24: **for** $i = 1 : n$ **do**
25: **if** $L_i^{\Upsilon} < E_i^{Target}$ **then**
26: **return** (Infeasible)
27: **return** $(\{b_i^j\}) \forall i, j : 1 \leq i \leq n, 1 \leq j \leq \Upsilon$

is designed to remove those constraints, since the objective is no more maximizing the total energy, but rather maximizing minimum energy.

The *Aggressive* scheme (Algorithm 3) chooses its baseline as the upper bound of the modulation level. In other words, it allocates the nodes with the maximum communication speed first, hence the name *Aggressive*. In each epoch, the algorithm iteratively looks for the node with the minimum remaining battery level before scaling down the modulation level for that node (line 11). The modulation level of a given node may be reduced multiple times in an epoch, as long as it remains as the node with the minimum remaining energy through the iterations. If the node hits the minimum modulation level limitation or the remaining slack is not enough for it to slow down by one modulation level, it is removed from the queue Q that keeps the nodes eligible for scaling in that epoch (lines 14 and 19). One advantage of this approach is that the energy-poor nodes are given more opportunities for reducing their modulation levels. At the end of each epoch, the algorithm checks for the minimum battery level constraint and reports infeasible if the constraint is not met (lines 22, 23). The final epoch has an extra step of checking the energy neutrality constraint (lines 25, 26).

The loops in Algorithm 3 have a fixed number of operations, except for the *while* loop in Line 10, which allows for at most $n \cdot B$ rounds of scaling. This is because there are initially n nodes in the queue. If scaling a candidate node is not feasible, it is immediately removed from the queue (Line 20). Nodes are also removed from the queue when they are assigned the minimum modulation level (Line 18). Otherwise, the node's transmission speed is scaled down by one level and the slack time gets updated

(Line 14). Therefore, in each round of the *while* loop, a node is either removed from the queue or scaled down. Nodes' modulation levels at most B times and there are n nodes. Therefore, the *while* loop terminates after at most $B \cdot n$ iterations.

The algorithm can be implemented using a priority queue. Each iteration of *Aggressive* includes an initial sort of the remaining battery levels assuming the baseline modulation level with the complexity of $O(n\log(n))$. Each node can be extracted and put back to the queue of eligible nodes for that epoch at most B times. The operations of extracting the minimum and inserting take $O(\log(n))$ time. Adding up the previous operations, the complexity of an iteration will be $O(n\log(n)+B\cdot n\cdot\log(n)) = O(B\cdot n\cdot\log(n))$. There are a total of Υ epochs and therefore iterations, resulting in an overall complexity of $O(\Upsilon \cdot B \cdot n \cdot \log(n))$. Since B is a small constant, the complexity of *Aggressive* can be simplified as $O(\Upsilon \cdot n \cdot \log(n))$.

5. EXPERIMENTAL EVALUATION

In order to evaluate the performance of the different algorithms for our optimization objectives under a wide variety of parameters, we implemented a discrete-event simulator in Matlab. We also used the IBM Cplex C++ library to solve the mixed integer programming problem instances optimally.

For DMS, we assumed Quadrature Amplitude Modulation (QAM) and adopted the radio parameters from Schurgers et al. [2003] by assigning $C_s = 12\cdot10^{-9}$, $C_e = 15\cdot10^{-9}$, and the symbol rate of $R_s = 62{,}500Hz$. Nodes communicate via packets of size $\rho = 128$ bytes. Each node transmits two packets within a single super-frame. Unless otherwise stated, the super-frame length/deadline is set to $D = 47.5ms$. There are five available modulation levels, selected from the set $\{2, 4, 6, 8, 10\}$.

The cluster is assumed to have $n = 8$ nodes, organized in a star topology. For energy harvesting, we used the solar profile trace measurements from Renner et al. [2009]. The measurement was obtained by sampling solar radiation at intervals of 30 seconds during a 2-month period in Hamburg, Germany. The daily energy profile is divided into $\Upsilon = 48$ epochs; hence, the length of each epoch is $c = 30$ minutes. The default number of super-frames per epoch is $\gamma = 20{,}000$. In a cluster with nodes distributed across an area, nodes may not have completely homogeneous solar energy collection profiles; for example, some of them may be subject to temporary blockage (due to clouds or moving objects). Hence, the daily solar radiation patterns observed by different nodes are derived also from the data available in Renner et al. [2009]. A sample daily energy harvesting profile for eight nodes is illustrated in Figure 4. Energy values in the plot are normalized with respect to the maximum energy harvested by a single node in the entire network during any epoch.

Following Jiang et al. [2005], we chose a super-capacitor of size $J_i = J = 500$ Joules for all nodes. Unless stated otherwise, the maximum node-level initial energy in our experiments is set to half of the battery capacity, that is, $\frac{J_i}{2}$. The actual initial energy level of each node, E_i^0, is a uniformly chosen random value between $\frac{J_i}{4}$ and $\frac{J_i}{2}$. Hence, the initial energy level of the nodes varies by a factor up to 50%. In all our experiments, we set the target energy $E_i^{Target} = \frac{J_i}{2} \geq E_i^0$, thereby enforcing the energy neutrality for all nodes. The energy values reported in the plots are normalized with respect to the maximum storage capacity. Each reported data value represents the average of 35 runs. For cases when a scheme fails to generate a feasible solution (i.e., violating either the deadline or the energy neutrality condition), no energy value is reported for that data point.

5.1. Evaluation of the Schemes That Maximize Total Remaining Energy

We first evaluate the performance of the algorithms that are proposed for maximizing the total remaining energy at the end of last epoch while satisfying the deadline and

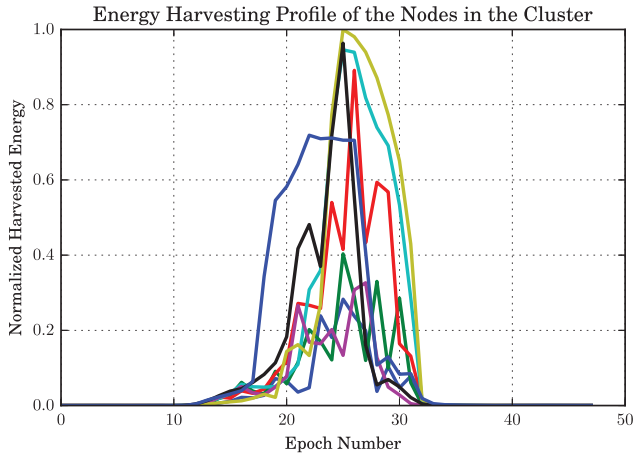


Fig. 4. Energy harvesting profiles.

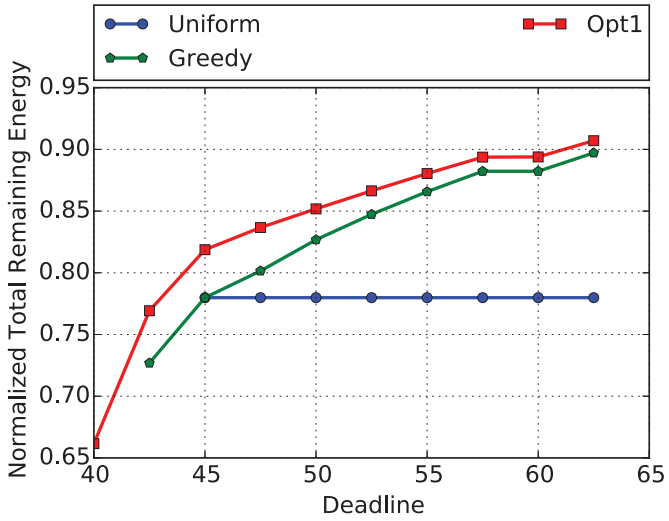
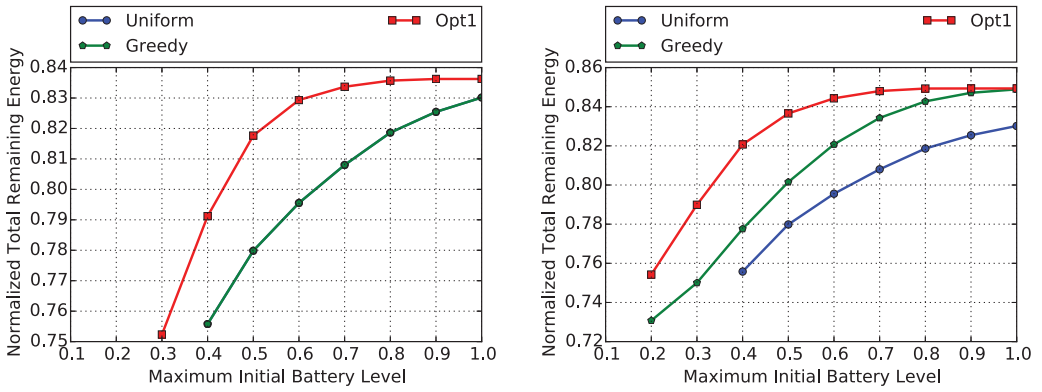


Fig. 5. Effect of deadline.

energy neutrality conditions. Specifically, we compare the performance of our heuristics *Greedy* and *Uniform* to that of the optimal but computationally expensive scheme *Opt1*. The reported energy values are normalized with respect to the collective battery capacity of the cluster (i.e., $n \cdot J_i = n \cdot J$).

The impact of the super-frame size: We first analyze the impact of the super-frame size (deadline) on the performance. Figure 5 shows the normalized energy levels for different algorithms. In general, larger super-frame sizes produce higher levels of remaining energy. This is because large super-frame sizes imply larger deadlines, which enable the system to use slower transmission speeds for individual nodes using DMS, leading to more energy savings. Notice that *Uniform* and *Greedy* are not able to find a feasible solution for deadlines smaller than 45*ms* and 42.5*ms*, respectively. Above those thresholds, *Greedy* performs close to the optimal solution *Opt1*, while the simpler *Uniform* achieved energy values within 13% of the other two in the worst case. We also



(a) a. $D = 43ms$ (b) b. $D = 47.5ms$

Fig. 6. Effect of initial battery level.

observe the increasing gap between *Uniform* and the other two schemes as we increase the deadline. In fact, the performance of *Uniform* remains constant in this experiment since the extra slack provided by increasing the deadline is not large enough to allow all the nodes to scale down their modulations collectively by one level. While *Opt1* and *Greedy* take advantage of extra slack to apply DMS on individual nodes, this is not possible for the simple *Uniform* modulation assignment algorithm.

The impact of the initial battery level: Our next experiment was designed to study the impact of the initial battery level on the remaining energy for two different values of deadline (Figure 6). We changed the maximum initial energy of the batteries from 10% to 100% of the battery capacity J_i. As we increase the maximum initial energy level, the total remaining energy also increases. However, the gains diminish with further increase in the initial energy levels. This is because the extra energy obtained by harvesting cannot be stored when the initial energy is already high, and the harvested energy is more than the energy consumed during the day. At the same time, the closing epochs of the period (the evening times) do not bring significant solar radiation (see Figure 4). The nodes rely on the energy stored in their batteries during those epochs. Consequently, the total remaining energy is restricted by the battery size and the number of "dark" epochs toward the end of the day. For $D = 43ms$ in Figure 6(a), *Uniform* and *Greedy* produce the same result since the tight deadline does not allow *Greedy* to reduce the modulation levels on individual nodes. But for larger deadlines, (e.g., when $D = 47.5ms$ in Figure 6(b)), *Greedy* is able to exploit the additional slack for individual nodes and yield more energy.

The impact of the harvesting power: We studied the impact of the available harvesting panel power in the next set of experiments. Specifically, we scaled the amplitude of the harvested power with a factor in the range [0.6, 1.1] and repeated the experiments (Figure 7). We observe that the performance of *Opt1* changes almost linearly with the scaling factor. *Uniform* and *Greedy* fail to find a feasible solution below the scaling factors of 0.85 and 0.7, respectively. It is worthwhile to note that due to the relatively small magnitude of the harvesting power figures (compared to the battery capacity), the changes in the total remaining energy are rather small even when we scale up and down the harvesting power figures.

The impact of the battery size: In this experiment (Figure 8), we scaled the maximum battery capacity to investigate the impact of the battery size. Specifically, J_i is scaled

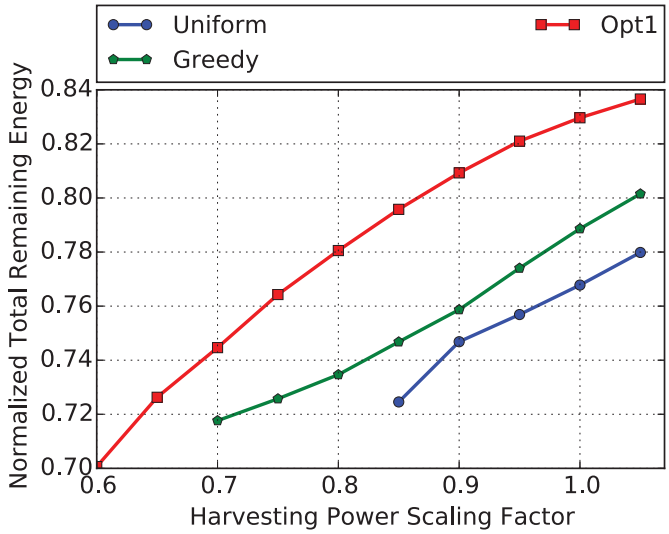


Fig. 7. Effect of harvesting power.

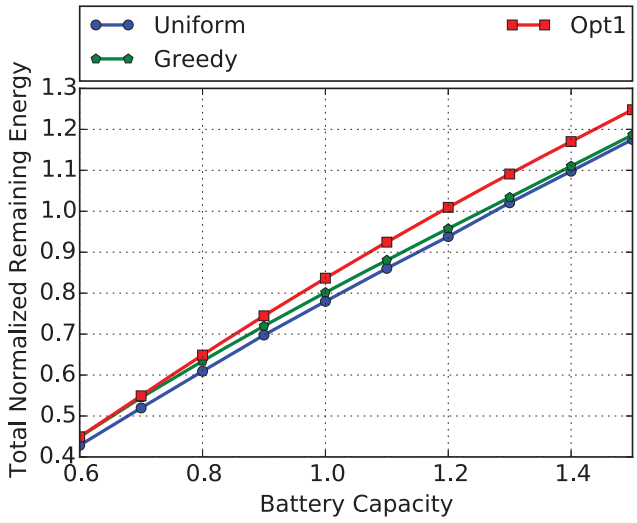


Fig. 8. Effect of battery size.

by a constant factor k, where $0.6 \leq k \leq 1.5$. In these experiments, with the battery capacity, we also change the values of the initial energy and the target energy so that their relationships with the battery capacity still hold. As the capacity of the battery increases, the initial energy of the nodes increases, and this has a positive impact on the total remaining energy since the consumed and harvested energy figures do not change. The apparent small difference between the schemes is due to the rather large range of the total normalized energy, as represented on the vertical axis of Figure 8.

The impact of the number of super-frames per epoch: The last experiment studies the effect of the number of super-frames per epoch, γ. As we increase γ, the total energy

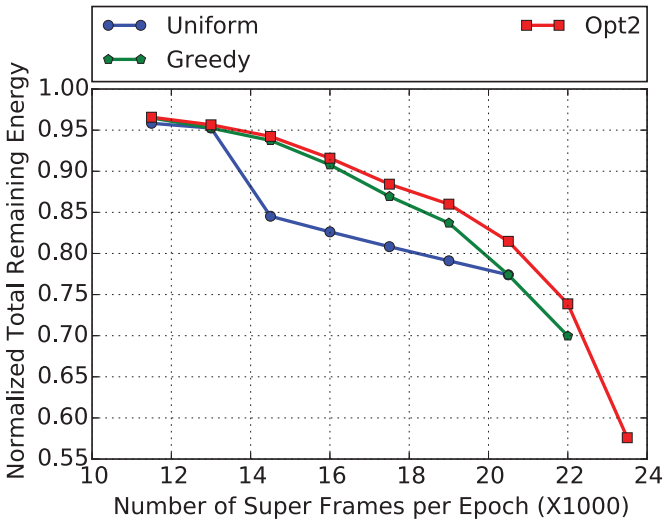


Fig. 9. Effect of the number of super-frames per epoch.

consumption due to the communication also increases, decreasing the remaining total energy as shown in Figure 9. An interesting observation is the super-linearly decreasing nature of the remaining energy, with increasing γ. This is because, when we increase the number of super-frames per epoch, the modulation levels previously calculated for the base case become suboptimal, and many nodes need to reduce their modulation levels to guarantee the deadline and/or energy neutrality. Because of the convex relationship between the modulation levels and energy consumption, a steep decrease in total energy is observed. Note that *Uniform* and *Greedy* fail to find a feasible solution when $\gamma \geq 20{,}500$ and $\gamma \geq 22{,}000$, respectively.

5.2. Evaluation of the Schemes That Maximize the Minimum Remaining Energy Across the Nodes

Our next set of experiments study the performance of the algorithms suggested for the objective of maximizing the minimum remaining energy across the entire cluster. Specifically, we compare the performance of *Uniform* and *Aggressive* algorithms to that of the optimal (but computationally expensive) scheme *Opt*2. In this section, the minimum energy observed on any node at the end of the last epoch is normalized with respect to the node-level battery capacity, $J_i = J$.

The impact of the super-frame size: Figure 10 shows the impact of increasing deadline on the performance of the algorithms. Again, the larger the super-frame is, the more opportunities there are to use DMS, and the remaining energy for all nodes (hence, also the minimum one) increases. *Aggressive* exhibits a lower performance compared to *Opt2*, but their performance difference declines with increasing super-frame size (from 30% to 8%).

A comparison between the results of this experiment and the equivalent experiment with the first objective (Figure 5) shows that the gap between the optimal algorithm and heuristics is greater for the objective of maximizing the minimum energy across all the nodes (Objective 2). This follows from the nature of the second objective in which a slightly lower single final energy level in a heuristic has a disproportionate impact—while such a decrease may be compensated by the increase in other

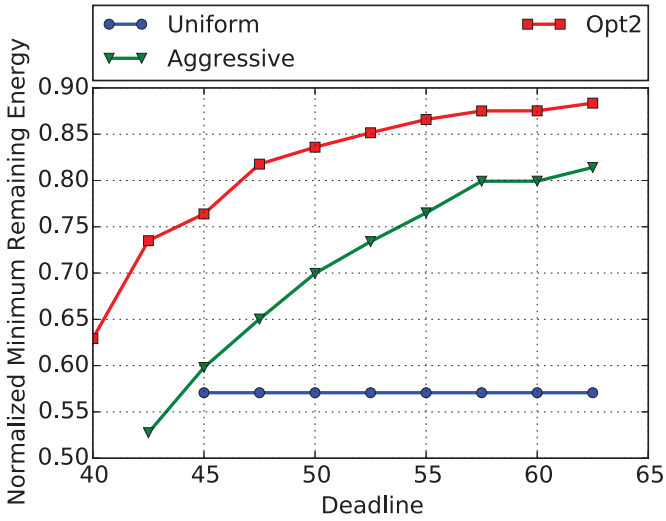
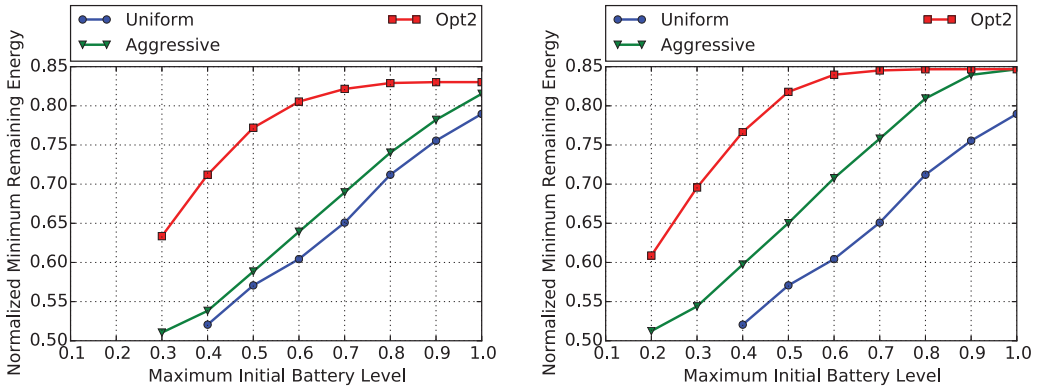


Fig. 10. Effect of deadline.



(a) a. $D = 43ms$ (b) b. $D = 47.5ms$

Fig. 11. Effect of initial battery level.

nodes' energy levels when the objective is to maximize the total remaining energy (Objective 1).

The impact of the initial battery level: Our next experiment shows the impact of the maximum initial battery level on the performance of the algorithms for two different deadlines (Figure 11). As expected, with larger deadlines, we get larger slack and better modulation scaling opportunities and this improves the performance. When the deadline is tight (Figure 11(a)), *Aggressive* is not able to find many opportunities to apply DMS and so performs close to *Uniform*, while it performs closer to the optimal algorithm *Opt2* when more slack time is provided (Figure 11(b)). *Aggressive* approaches *Opt2* as we increase the initial battery level in both cases.

Comparing the results in Figure 11 to those that depicted the relationship of the deadline to the total remaining energy (Figure 6), we observe that *Aggressive* is able to

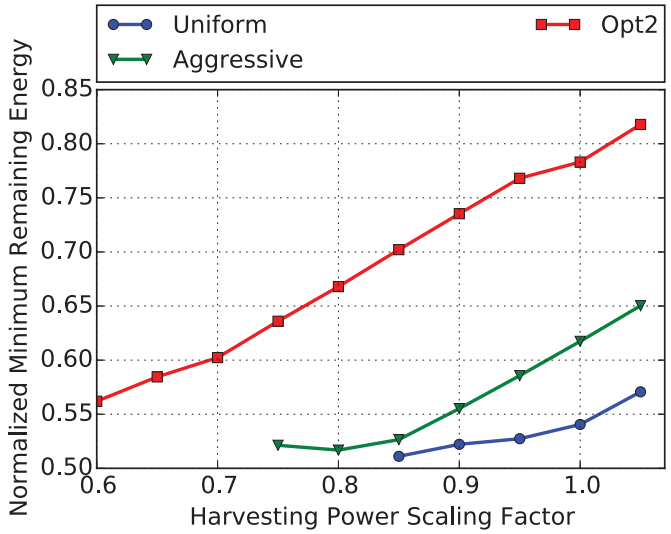


Fig. 12. Effect of the harvesting power.

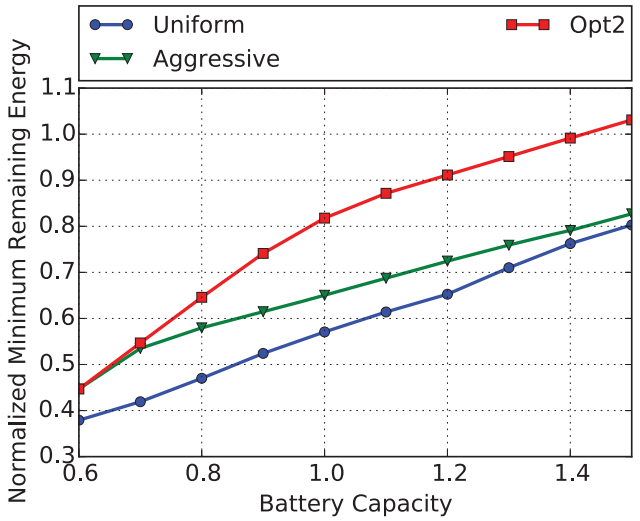


Fig. 13. Effect of the battery size.

outperform *Uniform* for both deadlines. Moreover, *Uniform* lags behind further when the deadline is larger, as it cannot apply different modulation levels to different nodes.

The impact of the harvesting power: Figure 12 shows the impact of the amount of harvesting power on the minimum remaining energy. As the harvesting power increases, the nodes can save more energy, therefore increasing the minimum remaining energy. Notice that *Aggressive* offers a better performance than *Uniform*; in fact, the latter is not able to yield feasible solutions for the harvesting power scaling factors less than 0.75.

The impact of the battery size: Minimum remaining energy increases almost linearly with the increasing battery capacity as shown in Figure 13. As the size of the battery is increased, more energy buffer becomes available for the nodes to prevent overflow

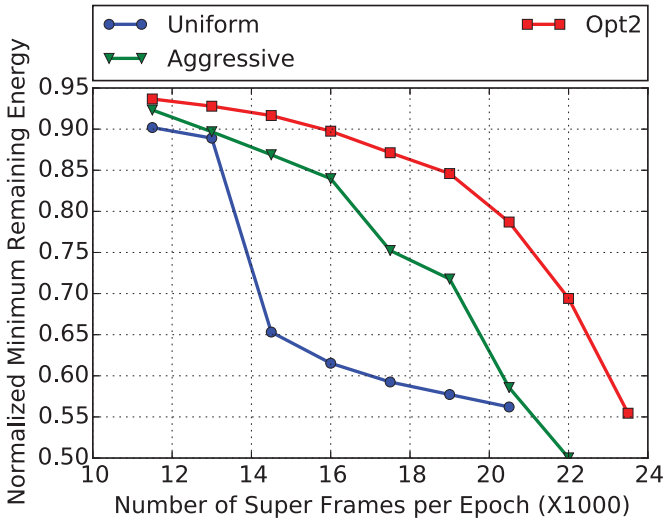


Fig. 14. Effect of the number of super-frames per epoch.

Table I. Runtime Comparison of Different Algorithms (Seconds)

Opt1	Opt2	Uniform	Greedy	Aggressive
1285.496	1388.4932	0.0018	0.0628	0.0707

of the excess harvested energy, and more feasible solutions with better performance values are obtained. This is particularly beneficial when more storage capacity is added to small batteries (low-capacity numbers). As the harvesting power is constant, the marginal return of such increases tends to diminish as we further increase the battery capacity.

The impact of the number of super-frames per epoch: Figure 14 shows the impact of increasing the cluster's workload in terms of the number of super-frames in an epoch. Similar to the results presented for Objective 1 (in Figure 9), the increasing workload causes steep decreases in the performance, because of the need to use higher modulation levels to guarantee feasibility, and the convex relationship between the energy consumption and the modulation levels. Once again, *Uniform*'s performance quickly degrades, and it is not able to generate feasible solutions when the number of super-frames per epoch exceeds 20,500.

The running times of the algorithms: The asymptotic complexity analysis of the proposed algorithms was presented in their corresponding subsections. We also ran experiments to measure the actual running times of the algorithms on an 2.6GHz dual-core iMac machine with 8GB memory, and the results are presented in Table I. The reported running times are obtained during the experiments that investigated the impact of the initial battery level. The results show the average running time for 350 experiments with different initial battery levels.

6. CONCLUSIONS

In this article, we considered cluster-based energy harvesting wireless sensor networks with timeliness requirements. In this framework, the nodes have to transmit their readings to the cluster head by a specific deadline to preserve application functionality. We considered the impact of Dynamic Modulation Scaling on these systems

for two different objectives. The first objective is to maximize the energy reserves while meeting the timing constraints under the energy neutrality conditions, which requires that consumed energy never exceeds the sum of harvested and stored energy during operation. The second optimization objective we considered is the maximization of the minimum energy across all nodes, to make sure that the energy levels are more evenly balanced in the network while still satisfying the timeliness and energy neutrality requirements. We formally showed that both problems can be encoded as instances of the Mixed Integer Linear Programming problem. We also proposed several fast heuristics that traded off optimality for a reduction in computational complexity. Using a real solar trace, we then compared the performance of the algorithms for both performance objectives. Our results indicate that, while very fast, the uniform modulation level assignment algorithm performs relatively poorly. However, the heuristic algorithms that allow individual nodes to adopt different slow-down factors in different epochs perform much better, and they approach the performance of the computationally expensive optimal algorithms, especially for large deadline or large initial battery level cases.

APPENDIX: DETAILS OF THE MIXED INTEGER PROGRAMMING FORMULATION

In this appendix, we provide the details of the Mixed Integer Linear Programming formulation given in Section 4.1, which is the basis for the optimal solution.

Objective Function

Our objective function is to maximize the overall remaining energy levels of all nodes at the end of last epoch (epoch number Υ).

$$Maximize \qquad \sum_{i=1}^{n} L_i^{\Upsilon}$$

A closed-form expression for the epoch-end energy levels (L_i^j values) requires expressing the energy consumption of each node as a function of the assigned modulation level. Specifically, a node requires $e(b)$ amount of energy to send M packets of size ρ bits over the channel in one epoch, when using the modulation level b:

$$e(b) = \gamma \cdot M \cdot \rho \cdot \frac{C_s(2^b - 1) + C_e}{b}.$$

Let us introduce a binary indicator variable α_{il}^j to indicate the corresponding modulation level choices from the set $\{b_1, \ldots, b_B\}$ for node i during epoch j. Specifically, α_{il}^j is set to 1 if and only if the l^{th} modulation level was selected for node i in epoch j; otherwise, it is 0. Then, the energy consumption of a node i, in a given epoch j, can be expressed as

$$e_i^j = \alpha_{i1}^j \cdot e(b_1) + \alpha_{i2}^j \cdot e(b_2) + \cdots + \alpha_{il}^j \cdot e(b_l) + \cdots + \alpha_{iB}^j \cdot e(b_B). \tag{10}$$

We add a constraint to show that exactly one of the B options for modulation level will be chosen for node i in epoch j:

$$\sum_{l=1}^{B} \alpha_{il}^j = 1, \ \forall i, j : \ 1 \le i \le n, 1 \le j \le \Upsilon.$$

Epoch Energy Levels

For any node, the available energy at the end of each epoch equals available energy at the end of the preceding epoch plus the difference between the harvested and consumed energy amounts during that epoch. Considering that node i cannot store more than

J_i units of energy, the remaining energy level at node i at the end of epoch j may be obtained as

$$L_i^j = \min\{J_i, L_i^{j-1} + c \cdot P_i^j - e_i^j\}.$$

The overflow energy that may not be stored in the battery of node i at the end of epoch j due to the capacity limits can be expressed by variable θ_i^j (the excess energy that cannot be stored and that is dissipated as heat). Overflow variables are nonnegative real numbers: $\theta_i^j \geq 0$. We can write

$$\begin{aligned} L_i^j &= L_i^{j-1} + c \cdot P_i^j - e_i^j - \theta_i^j \\ &= E_i^0 + \sum_{k=1}^{j} \left(c \cdot P_i^k - e\left(b_i^k\right) - \theta_i^k\right) \\ &= E_i^0 + \sum_{k=1}^{j} \left(c \cdot P_i^k\right) - \frac{\gamma \cdot M \cdot \rho}{b_i^k}\left(C_s \cdot \left(2^{b_i^k} - 1\right) + C_e\right) - \theta_i^k. \end{aligned}$$

Then the objective function can be rewritten as

$$\textit{Maximize} \qquad \sum_{i=1}^{n} E_i^0 + \sum_{k=1}^{\Upsilon} \left(c \cdot P_i^k - e_i^k - \theta_i^k\right).$$

By substituting the value of e_i^j from Equation (10), we obtain

$$\textit{Maximize} \qquad \sum_{i=1}^{n} E_i^0 + \sum_{k=1}^{\Upsilon} \left(c \cdot P_i^k - \frac{\gamma \cdot M \cdot \rho}{b_i^k}\left(C_s \cdot \left(2^{b_i^k} - 1\right) + C_e\right) - \theta_i^k\right).$$

Since E_i^0 and $c \cdot P_i^k$ are not a function of the modulation level, this is equivalent to

$$\textit{Minimize} \qquad \sum_{i=1}^{n} \sum_{k=1}^{\Upsilon} \left(\frac{\gamma \cdot M \cdot \rho}{b_i^k}\left(C_s \cdot \left(2^{b_i^k} - 1\right) + C_e\right) + \theta_i^k\right). \tag{11}$$

By using our binary indicator variables $\{\alpha_{il}^k\}$, we can rewrite as

$$\textit{Minimize} \qquad \sum_{i=1}^{n} \sum_{k=1}^{\Upsilon} \sum_{l=1}^{B} \left(\frac{\alpha_{il}^k \cdot \gamma \cdot M \cdot \rho}{b_l}\left(C_s \cdot \left(2^{b_l} - 1\right) + C_e\right) + \theta_i^k\right). \tag{12}$$

Time Constraints
The real-time characteristic of the application requires all nodes to complete their transmission within the deadline of each super-frame, D:

$$\sum_{i=1}^{n} \frac{t_i^j}{\gamma} = \sum_{i=1}^{n} \left(\frac{M \cdot \rho}{R_s \cdot b_i^j}\right) \leq D, \qquad \forall\, 1 \leq j \leq \Upsilon. \tag{13}$$

Energy Constraints
The energy consumption characteristics must prevent both battery overflow and underflow conditions while guaranteeing energy neutrality. The prevention of battery underflow ensures that the battery level never drops to zero. We may combine the

target energy condition with the battery underflow condition by defining a variable ζ_i^j that shows the minimum allowable battery level of node i at the end of epoch j.

Clearly, we have $\zeta_i^j = 0,\ 1 \le j \le \Upsilon - 1$, and $\zeta_i^{\Upsilon} = E_i^{Target} = E_i^0$:

$$L_i^j = E_i^0 + \sum_{k=1}^{j} \left(c \cdot P_i^k - \frac{\gamma \cdot M \cdot \rho}{b_i^k} \left(C_s \cdot \left(2^{b_i^k} - 1 \right) + C_e \right) - \theta_i^k \right) \ge \zeta_i^j,$$
$$\forall i, j : 1 \le i \le n, 1 \le j \le \Upsilon.$$

Preventing battery overflow ensures that the excess energy that cannot be stored in the battery is not taken into account to guarantee the energy constraints of the subsequent epochs. In other words, the battery level at the end of each super-frame should not exceed the node's battery capacity. Combining all these constraints, we get

$$\zeta_i^j \le E_i^0 + \sum_{k=1}^{j} \left(c \cdot P_i^k - \frac{\gamma \cdot M \cdot \rho}{b_i^k} \left(C_s \cdot \left(2^{b_i^k} - 1 \right) + C_e \right) - \theta_i^k \right) \le J_i$$
$$\forall i, j : 1 \le i \le n, 1 \le j \le \Upsilon.$$

Putting all the constraints together, we obtain the following mixed integer programming formulation:

$$\underset{\theta,\alpha}{\text{Minimize}} \quad \sum_{i=1}^{n} \sum_{k=1}^{\Upsilon} \sum_{l=1}^{B} \left(\frac{\alpha_{il}^k \cdot \gamma \cdot M \cdot \rho}{b_l} \left(C_s \cdot (2^{b_l} - 1) + C_e \right) + \theta_i^k \right)$$

$$\text{Subject to} \quad \sum_{i=1}^{n} \sum_{l=1}^{B} \left(\frac{\alpha_{il}^j \cdot M \cdot \rho}{R_s \cdot b_l} \right) \le D, \qquad \forall j : 1 \le j \le \Upsilon$$

$$\zeta_i^j \le E_i^0 + \sum_{k=1}^{j} \left(c \cdot P_i^k - \sum_{l=1}^{B} \left(\frac{\alpha_{il}^k \cdot \gamma \cdot M \cdot \rho}{b_l} \left(C_s \cdot (2^{b_l} - 1) + C_e \right) \right) - \theta_i^k \right) \le J_i,$$
$$\forall i, j : 1 \le i \le n, 1 \le j \le \Upsilon$$

$$\alpha_{il}^j \in \{0, 1\}, \qquad \forall i, j, l : 1 \le i \le n, 1 \le j \le \Upsilon, 1 \le l \le B$$

$$\sum_{l=1}^{B} \alpha_{il}^j = 1, \qquad \forall i, j : 1 \le i \le n, 1 \le j \le \Upsilon$$

$$\theta_i^j \ge 0, \qquad \forall i, j : 1 \le i \le n, 1 \le j \le \Upsilon.$$

REFERENCES

Novella Bartolini, Tiziana Calamoneri, Tom La Porta, Chiara Petrioli, and Simone Silvestri. 2012. Sensor activation and radius adaptation (SARA) in heterogeneous sensor networks. *ACM Transactions on Sensor Networks* 8, 3, Article 24 (Aug. 2012), 34 pages. DOI: http://dx.doi.org/10.1145/2240092.2240098

CC1100. 2014. Low-Power Sub-1 GHz RF Transceiver. Retrieved from http://www.ti.com/lit/ds/swrs038d/swrs038d.pdf. (2014).

CC2500. 2014. Low-Cost Low-Power 2.4 GHz RF Transceiver. Retrieved from http://www.ti.com/lit/ds/symlink/cc2500.pdf.

Shuguang Cui, Andrea J. Goldsmith, and Ahmad Bahai. 2005. Energy-constrained modulation optimization. *IEEE Transactions on Wireless Communications* 4, 5 (2005), 2349–2360.

Yuemin Ding and Seung Ho Hong. 2013. CFP scheduling for real-time service and energy efficiency in the industrial applications of IEEE 802.15.4. *Journal of Communications and Networks* 15, 1 (2013), 87–101.

Melike Erol-Kantarci and Hussein T. Mouftah. 2012. Suresense: Sustainable wireless rechargeable sensor networks for the smart grid. *IEEE Wireless Communications* 19, 3 (2012), 30–36.

Benazir Fateh and Manimaran Govindarasu. 2013. Energy minimization by exploiting data redundancy in real-time wireless sensor networks. *Ad Hoc Networks* 11, 6 (2013), 1715–1731.

B. Fateh and M. Govindarasu. 2015. Joint scheduling of tasks and messages for energy minimization in interference-aware real-time sensor networks. *IEEE Transactions on Mobile Computing* 14, 1 (Jan. 2015), 86–98. DOI: http://dx.doi.org/10.1109/TMC.2013.81

P. M. Glatz, L. B. Hormann, C. Steger, and R. Weiss. 2011. HANS: Harvesting aware networking service for energy management in wireless sensor networks. In *Proceedings of the 18th International Conference on Telecommunications*. DOI: http://dx.doi.org/10.1109/CTS.2011.5898914

Maria Gorlatova, Peter Kinget, Ioannis Kymissis, Dan Rubenstein, Xiaodong Wang, and Gil Zussman. 2010. Energy harvesting active networked tags (EnHANTs) for ubiquitous object networking. *IEEE Wireless Communications* 17, 6 (2010), 18–25.

Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. 2013. Internet of things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems* 29, 7 (2013), 1645–1660.

Vehbi C. Gungor and Gerhard P. Hancke. 2009. Industrial wireless sensor networks: Challenges, design principles, and technical approaches. *IEEE Transactions on Industrial Electronics* 56, 10 (2009), 4258–4265.

Zdeněk Hanzálek and Petr Jurčík. 2010. Energy efficient scheduling for cluster-tree wireless sensor networks with time-bounded data flows: Application to IEEE 802.15.4/ZigBee. *IEEE Transactions on Industrial Informatics* 6, 3 (2010), 438–450.

F. Iannello, O. Simeone, and U. Spagnolini. 2012. Medium access control protocols for wireless sensor networks with energy harvesting. *IEEE Transactions on Communications* 60, 5 (May 2012), 1381–1389. DOI: http://dx.doi.org/10.1109/TCOMM.2012.030712.110089

Xiaofan Jiang, Joseph Polastre, and David Culler. 2005. Perpetual environmentally powered sensor networks. In *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, 2005 (IPSN'05)*. IEEE, 463–468.

Aman Kansal, Jason Hsu, Sadaf Zahedi, and Mani B. Srivastava. 2007. Power management in energy harvesting sensor networks. *ACM Transactions on Embedded Computing Systems* 6, 4, Article 32 (2007). DOI: http://dx.doi.org/10.1145/1274858.1274870

Aman Kansal and Mani B. Srivastava. 2003. An environmental energy harvesting framework for sensor networks. In *Proceedings of the International Symposium on Low Power Electronics and Design*. ACM, 6. DOI: http://dx.doi.org/10.1145/871506.871624

K. Kinoshita, T. Okazaki, H. Tode, and K. Murakami. 2008. A data gathering scheme for environmental energy-based wireless sensor networks. In *Proceedings of the 5th IEEE Conference on Consumer Communications and Networking*. DOI: http://dx.doi.org/10.1109/ccnc08.2007.165

Feng Li, Jun Luo, Gaotao Shi, and Ying He. 2013. FAVOR: Frequency allocation for versatile occupancy of spectrum in wireless sensor networks. In *Proceedings of the 14th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 39–48.

Ye Li, Bertan Bakkaloglu, and Chaitali Chakrabarti. 2007. A system level energy model and energy-quality evaluation for integrated transceiver front-ends. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 15, 1 (2007), 90–103.

Shixin Luo, Rui Zhang, and Teng Joon Lim. 2013. Optimal save-then-transmit protocol for energy harvesting wireless transmitters. *IEEE Transactions on Wireless Communications* 12, 3 (March 2013), 1196–1207. DOI: http://dx.doi.org/10.1109/TWC.2013.012413.120488

Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. 2002. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*. ACM, 88–97.

Basilis Mamalis, Damianos Gavalas, Charalampos Konstantopoulos, and Grammati Pantziou. 2009. Clustering in wireless sensor networks. *RFID and Sensor Networks: Architectures, Protocols, Security and Integrations* (2009). CRC Press, 324–353.

B. Medepally and N. B. Mehta. 2010. Voluntary energy harvesting relays and selection in cooperative wireless networks. *IEEE Transactions on Wireless Communications* 9, 11 (2010), 3543–3553. DOI: http://dx.doi.org/10.1109/TWC.2010.091510.100447

Guowang Miao, Nageen Himayat, and Geoffrey Ye Li. 2010. Energy-efficient link adaptation in frequency-selective channels. *IEEE Transactions on Communications* 58, 2 (2010), 545–554.

N. Michelusi and M. Zorzi. 2013. Optimal random multiaccess in energy harvesting wireless sensor networks. In *Proceedings of the 2013 IEEE International Conference on Communications Workshops (ICC'13)*. 463–468. DOI:http://dx.doi.org/10.1109/ICCW.2013.6649278

N. Michelusi and M. Zorzi. 2015. Optimal adaptive random multiaccess in energy harvesting wireless sensor networks. *IEEE Transactions on Communications* 63, 4 (April 2015), 1355–1372. DOI:http://dx.doi.org/10.1109/TCOMM.2015.2402662

Clemens Moser, Jian-Jia Chen, and Lothar Thiele. 2008. An energy management framework for energy harvesting embedded systems. *ACM Journal on Emerging Technologies in Computing Systems* 6, 2, Article 7 (2008), 21 pages. DOI:http://dx.doi.org/10.1145/1773814.1773818

Dong Kun Noh, Lili Wang, Yong Yang, Hieu Khac Le, and Tarek Abdelzaher. 2009. Minimum variance energy allocation for a solar-powered sensor system. In *Distributed Computing in Sensor Systems*. Springer, 44–57.

Myung-Gon Park, Kang-Wook Kim, and Chang-Gun Lee. 2011. Holistic optimization of real-time IEEE 802.15.4/ZigBee networks. In *Proceedings of the IEEE International Conference on Advanced Information Networking and Applications*. DOI:http://dx.doi.org/10.1109/AINA.2011.15

Padmanabhan Pillai and Kang G. Shin. 2001. Real-time dynamic voltage scaling for low-power embedded operating systems. In *ACM SIGOPS Operating Systems Review*, Vol. 35. ACM, 89–102.

Tifenn Rault, Abdelmadjid Bouabdallah, and Yacine Challal. 2014. Energy efficiency in wireless sensor networks: A top-down survey. *Computer Networks* 67 (2014), 104–122. DOI:http://dx.doi.org/10.1016/j.comnet.2014.03.027

Christian Renner, Jürgen Jessen, and Volker Turau. 2009. Lifetime prediction for supercapacitor-powered wireless sensor nodes. In *Proceedings of the 8th GI/ITG KuVS Fachgespräch Drahtlose Sensornetze*.

Francesco Restuccia and Sajal K. Das. 2016. Optimizing the lifetime of sensor networks with uncontrollable mobile sinks and QoS constraints. *ACM Transactions on Sensor Networks (TOSN)* 12, 1 (2016), 2.

Curt Schurgers, Vijay Raghunathan, and Mani B. Srivastava. 2003. Power management for energy-aware communication systems. *ACM Transactions on Embedded Computing Systems* 2, 3 (2003), 431–447. DOI:http://dx.doi.org/10.1145/860176.860184

Sensus. 2009. iCon Advanced Security FlexNet Integrated Display Transceiver. https://apps.fcc.gov/els/GetAtt.html?id=109424.

Wei Shen, Tingting Zhang, F. Barac, and M. Gidlund. 2014. PriorityMAC: A priority-enhanced MAC protocol for critical traffic in industrial wireless sensor and actuator networks. *IEEE Transactions on Industrial Informatics* 10, 1 (Feb. 2014), 824–835. DOI:http://dx.doi.org/10.1109/TII.2013.2280081

IEEE Radio Standard. 2012. IEEE Standard for Local and Metropolitan Area Networks. http://standards.ieee.org/findstds/standard/802.15.4g-2012.html.

S. Sudevalayam and P. Kulkarni. 2011a. Energy harvesting sensor nodes: Survey and implications. *IEEE Communications Surveys Tutorials* 13, 3 (2011), 443–461. DOI:http://dx.doi.org/10.1109/SURV.2011.060710.00094

Sujesha Sudevalayam and Purushottam Kulkarni. 2011b. Energy harvesting sensor nodes: Survey and implications. *IEEE Communications Surveys & Tutorials* 13, 3 (2011), 443–461.

G. Sudha Anil Kumar, G. Manimaran, and Z. Wang. 2007. Energy-aware scheduling of real-time tasks in wireless networked embedded systems. In *Proceedings of the 28th IEEE International Real-Time Systems Symposium*. DOI:http://dx.doi.org/10.1109/RTSS.2007.21

Thales. 2015. Radio Communication Products. Retrieved from http://www.thalesgroup.com/sites/default/files/asset/document/radiocomm unication_products_solutions_2015_0.pdf.

Emanuele Toscano and Lucia Lo Bello. 2012. Multichannel superframe scheduling for IEEE 802.15.4 industrial wireless sensor networks. *IEEE Transactions on Industrial Informatics* 8, 2 (2012), 337–350.

Ambuj Varshney, Luca Mottola, Mats Carlsson, and Thiemo Voigt. 2015. Directional transmissions and receptions for high-throughput bulk forwarding in wireless sensor networks. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. ACM, 351–364.

Christopher M. Vigorito, Deepak Ganesan, and Andrew G. Barto. 2007. Adaptive control of duty cycling in energy-harvesting wireless sensor networks. In *Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, 2007 (SECON'07)*. IEEE, 21–30.

Liguang Xie, Yi Shi, Y. Thomas Hou, Wenjing Lou, and Hanif D. Sherali. 2013. On traveling path and related problems for a mobile station in a rechargeable sensor network. In *Proceedings of the 14th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 109–118.

Bo Zhang, Robert Simon, and Hakan Aydin. 2013. Harvesting-aware energy management for time-critical wireless sensor networks with joint voltage and modulation scaling. *IEEE Transactions on Industrial Informatics* 9, 1 (2013), 514–526.

Fan Zhang and Samuel T. Chanson. 2005. Improving communication energy efficiency in wireless networks powered by renewable energy sources. *IEEE Transactions on Vehicular Technology* 54, 6 (2005), 2125–2136.

Runwei Zhang, Patrick Thiran, and Martin Vetterli. 2015. Virtually moving base stations for energy efficiency in wireless sensor networks. In *Proceedings of the 16th ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 357–366.

Zhong Zhou, Shengli Zhou, Shuguang Cui, and Jun-Hong Cui. 2008. Energy-efficient cooperative communication in a clustered wireless sensor network. *IEEE Transactions on Vehicular Technology* 57, 6 (2008), 3618–3628.

Received October 2015; revised August 2016; accepted September 2016