On Self-Configuring IoT With Dual Radios: A Cross-Layer Approach

Jinhwan Jung, Joonki Hong, and Yung Yi

Abstract—Growing interest in emerging IoT applications provides a strong drive to release a plethora of communication radios from different standards, which are largely classified into short-range (IEEE 802.15.4) and long-range radios (IEEE 802.15.4g). In this paper, we propose a joint, self-configuring MAC and routing protocol, SEDA-Net, which aims at adaptively choosing the best configuration for communication coordination and data delivery, depending on different deployed topologies and external conditions. SEDA-Net is a combination of SEDA-MAC, SEDA-Routing, and Cross-Opt. SEDA-MAC and SEDA-Routing adaptively determine the best radio configuration for communication coordination under duty-cycling and each node's next-hop over which radio and Cross-Opt jointly optimizes inter-coupled MAC and routing in an iterative manner. SEDA-Net differs from prior approaches which are designed with static configurations of radios and/or mainly with the goal of throughput maximization for dual Wi-Fi or Wi-Fi/LTE setups. We implement SEDA-Net on Contiki OS and perform extensive simulations and experiments using a testbed in an office building. This testbed consists of 45 nodes equipped with a commercial platform, Firefly, having 2.4 GHz short-range and 920 MHz long-range radios. We demonstrate that energy efficiency quantified by the network lifetime increases by up to 2.1 times, compared to that of existing approaches.

Index Terms—Dual radios, Internet of Things, wireless sensor network, MAC protocol, routing protocol, cross-layer

1 Introduction

We are now experiencing the emergence of a plethora of new IoT applications in various domains, e.g., environmental monitoring, surveillance systems, and consumer electronics. They lead a strong drive for a surge of communication radios based on various communication standards with different characteristics. These radios are largely classified into IEEE 802.15.4 [1] at 2.4 GHz[1] and IEEE 802.15.4g [2] at sub-GHz, where the former is characterized by short communication range with high data rate, we call *short-range radio*, and the latter has long communication range with low data rate, we call *long-range radio*. We also have a list of products combining both standards with dual radios in the market for flexible operation. These are often called dual-radio motes, including Firefly [3], Waspmote [4], and OpenMote B [5], where we choose Firefly (Fig. 1b) in our testbed (see Section 5 for details).

Our focus is on a network of IoT nodes with such dual radios. The key design goals for network protocols in IoTs include energy efficiency and self-configuration so as to run systems without frequent battery replacement and cumbersome manual operation. The main advantage of configuring the IoT network with dual radios is that a few long-range connections can dramatically shorten the path among any two nodes in the network. In other words, well-configured dual-radio networks will have much shorter network distance – as the length of paths from data sources to a destination dramatically decreases, the energy cost for network operation (e.g., caused by long multi-hop delivery) will decrease accordingly.

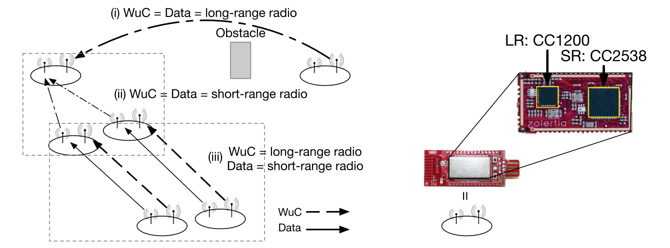
In terms of a dual-radio node, a popular approach to save energy is to switch off the radio as much as possible, depending on occasional or periodic data delivery. Two energy-saving directions exist as the state-of-the-art solutions. First, one can install an ultra-low-power (ULP) wake-up radio that is always awake and acts as a sentinel [6], [7], [8] while a main radio is switched off. This wake-up radio's role is to wake up its associated main radio for wake-up calls from other nodes, but it has limited RX sensitivity (i.e., short communication range) for energy efficiency. Second, one equips a more "powerful" radio than the ULP wake-up radio, but lets it duty-cycled. There exist an extensive array of efficient duty-cycling protocols proposed in literature [9], [10], [11], [12]. Under this design, a popular choice is to adopt a *static* configuration under which the long-range radio is used as the wake-up radio [13], [14], [15], [16], [17] – it helps to avoid multi-hop wake-up calls which may make the protocol highly complicated.

Despite the potential advantages of networks with dual radios, existing "rigid" solutions do not seem to come from fully exploring all design spaces, and thus are often sub-optimal. For example, in the second design mentioned earlier for dual radios, it is clear that statically and globally using the long-range radio for a wake-up radio may not be optimal in some cases, where which radio should be used for the wake-up radio can be even *location-dependent*. Thus, we allow a *heterogeneous radio configuration* as in Fig. 1a to be superior to other static and fixed ones, and such situation-

1. It is also known as ZigBee.

• *The authors are with the Department of Electrical Engineering, KAIST, Daejeon 34141, South Korea.*
E-mail: {jhjung, joonki}@lanada.kaist.ac.kr, yiyung@kaist.edu.

Manuscript received 28 Aug. 2019; revised 26 Feb. 2021; accepted 7 Mar. 2021. Date of publication 0 . 0000; date of current version 0 . 0000.
(Corresponding author: Yung Yi.)
Digital Object Identifier no. 10.1109/TMC.2021.3066441



(a) An example of heterogeneous radio configuration (b) Dual-radio mote in our testbed

Fig. 1. WuC: Wake-up Call. (a) An example configuration under a dual-radio setup, where a mixture of (i) long-range radio for both wake-up call and data, (ii) short-range radio for both wake-up call and data, and (iii) long-range radio for wake-up call and short-range radio for data is possible in one network. (b) Firefly [3]: a sensor board equipped with dual-radio.

dependent configuration is desired to be found in a self-organizing manner. However, it is far from being trivial to find a situation-dependent configuration in a self-organizing manner, since the best solution depends on various factors such as radio specification, application traffic pattern, node deployment, and even environmental factors (e.g., external interference). This makes the configuration problem coupled with duty-cycling (i.e., MAC layer) and routing (i.e., network layer), and thus the best configuration should be found in a large search space.

1.1 Contribution

In this paper, we explore dual radios' configuration spaces more widely and design a joint MAC and routing protocol to maximize the network lifetime as a primary metric, by finding the best one in a self-configuring and distributed manner. MAC is in charge of determining the wake-up radio for coordinating the communication for each TX-RX pair, and routing offers a rule of how a node chooses its next-hop for data delivery over the main radio. Thus, the best radio configuration over each TX-RX pair should involve the decisions of MAC and routing, thereby making a cross-layer design crucial. We summarize our main contributions in what follows.

- We propose ***SEDA-Net*** (SElf-DuAl-Net), which finds an energy-efficient configuration of dual radios and adapts itself to characteristics of dual radios, offered traffic pattern, and node deployment in a self-configuring and distributed manner. SEDA-Net is a generalization of existing fixed solutions in a single radio and dual radios with duty-cycling. To the best of our knowledge, this is the first joint MAC and routing protocol that considers dual radios and adaptively chooses the best radio configuration in a distributed manner to maximize the network lifetime. SEDA-Net is designed under the rationale that it is a heuristic approximation trying to solve an NP-complete optimization problem (see Section 3.2), which has a goal of maximizing network lifetime under the dual-radio setup.
- SEDA-Net consists of three sub-components: ***SEDA-MAC***, ***SEDA-Routing***, and ***Cross-Opt***. SEDA-MAC is a new MAC protocol that handles (i) how to choose the best radio for wake-up for each TX-RX pair and (ii) how to perform duty-cycling efficiently. SEDA-Routing constructs a routing topology, balancing the traffic well with the goal of maximizing the network lifetime. We use the RPL protocol [18], which is a *de facto* standard, as a base of SEDA-Routing. SEDA-MAC and SEDA-Routing are inter-connected by Cross-Opt that iteratively updates the parameters of SEDA-MAC and SEDA-Routing from a cross-layer perspective, so as to finally determine the appropriate configuration.
- We evaluate SEDA-Net by implementing it on the Contiki OS [19] for both simulations and real experiments. In simulations, we use the Cooja simulator which allows a variety of controllable setups and micro-benchmarks inside the Contiki OS. In real experiments, we build a testbed of 45 Firefly motes [3] (CC2538 for short and CC1200 for long-range radios) in a five-story office building. We observe that the average lifetime of SEDA-Net is up to 113 percent longer than that of other non-adaptive existing protocols, and SEDA-Net achieves delay reduction up to 49 percent and reliability improvement up to 77 percent over them.

2 Preliminary and Related Work

2.1 Preliminary

We first clarify the term *communication coordination*, which will be used as an important concept throughout this paper. We call communication coordination the process of preparing for TX and RX to be awake at the same time, so that they can exchange data. In single radio networks with duty-cycling, the radio is first used for communication coordination over the same radio, before exchanging data. When it is extended for dual radios, the wake-up radio should do such communication coordination, and the main radios at TX and RX are woken up by the wake-up radios, and then become ready for data delivery. In dual-radio with ULP (Ultra-Low-Power) radio, which is beyond our interest, communication coordination is relatively easy, because the wake-up radio of TX can express its intention of transmission to that of RX at any time.

As a routing protocol for low-power and lossy networks, RPL [18] was standardized by the IETF. RPL is a distance vector routing protocol that builds a destination-oriented directed acyclic graph (i.e., a tree). Each node, say v, exchanges routing information using DIO messages which include the routing information such as v's rank (i.e., "distance" to the destination). This information is used to join a tree and choose the best next-hop among its neighbors, following a given objective function. The objective function defines how to choose the next-hop. To be more precise, for a node v, the objective function typically chooses the next-hop u that has the minimum value of $\texttt{Rank}(u) + w_{vu}$ over all neighboring node u, where w_{vu} is the weight over the link vu. For example, OF0 [20] uses a hop count for rank and ETX (Expected Transmission Count) for weight. Thus, RPL can be generalized to various IoT networks by customizing the objective function accordingly.

2.2 Related Work

An extensive array of researches for duty-cycling is classified into many types under different criteria: synchronous versus asynchronous and sender-initiated versus receiver-initiated, having their own pros and cons, see a nice exhaustive survey in [12]. In dual-radio networks, different mechanisms for efficient duty-cycling of the wake-up radio have been proposed in [13], [14], [15], [16]. In [13], [14], they propose an efficient duty-cycling method to wake-up radio using energy consumption analysis. Other researches in [15], [16] are similar to ours in the sense that a dual-radio setup of 2.4 GHz short-range radio and sub-GHz long-range radio is utilized. However, their configuration is fixed to dual radios with duty-cycled wake-up radio, where the sub-GHz radio is used for the wake-up radio and the 2.4 GHz radio is for the data radio. Clearly, this restricts the potential benefit of allowing situation-dependent radio configurations. Recently, the authors in [21] propose a dual-radio network that separates control and data planes using different radios. They choose LoRa [22] to form one-hop control plane and use ZigBee-based data plane to collect data in a multi-hop manner. However, applying LoRa to the network requires a specialized controller which is much more expensive than other sub-GHz radios. A protocol for making a collection tree, utilizing both short- and long-range radios, has been proposed in [23]. This, however, does not consider the duty-cycling in MAC and thus the practical application can be limited under low-energy IoT applications.

At the routing layer, finding the best path from sensors to a base station is the most important function in terms of their target performance metrics (e.g., network lifetime, reliability, delay) in WSNs (Wireless Sensor Networks) and LLNs (Low power Lossy Networks). In [24], [25], [26], the authors theoretically formulate a network lifetime maximization problem, and they also propose heuristic algorithms because the problem is NP-Hard. Recently, to design practical routing algorithms, [27], [28], [29], [30] have proposed RPL-based protocols, where they build load-balanced routing structures by bounding the nodes' degree [27] or using the queue length [29]. However, no prior work exists for the dual-radio setup with duty-cycling from the cross-layer's perspective, jointly considering MAC and routing.

A vast collection of network protocols for multi-radio nodes have also been proposed in wireless mesh networks, see e.g., [31], [32], [33], [34], [35], [36], [37]. The protocols are designed to maximize throughput performance by bonding multiple Wi-Fi radios [31], [32], [33], or to improve energy efficiency by smartly switching between cellular (i.e., 3G or 4G) and Wi-Fi interfaces [34], [35]. The studies by [36], [37] propose channel allocation algorithms for multiple radios to minimize interference. The aforementioned works on multi-radio protocols significantly differ from this paper in the following sense: First, our design should consider communication coordination in dual radios with duty-cycling as a necessary protocol component. In LLNs, low power radios perform two roles: (a) communication coordination and (b) data delivery. Depending on numerous factors such as radio characteristics, deployed locations, and target metrics, one radio can be superior to another in terms of performing communication coordination or data delivery; thus the role of those radios should be chosen carefully, where in the aforementioned multi-radio protocols, data delivery has been the main interest. Second, such an intelligent allocation of roles should be done with the goal of maximizing energy efficiency, quantified by lifetime maximization in this paper, which is done in the cross-layer context by smartly choosing the roles of each radio (MAC) and a routing topology (Routing).

In WSNs and LLNs, cross-layer designs for energy efficiency have also been proposed [38], [39], [40], [41]. They have studied how WSNs operate to maximize energy efficiency by controlling transmission power [38] or clustering [41] along with packet size or the amount of data aggregation. There exist researches that jointly consider the physical layer with MAC or network layers. In [40], the authors proposed duty-cycling optimization while modulation levels are controlled in the physical layer. In [39], the authors proposed a routing protocol over the precisely modeled path loss by considering both small-scale and large-scale fading.

3 SEDA-NET: DESIGN OVERVIEW

3.1 Basic Modules and Goal

We first present the background of the baseline MAC and routing modules under SEDA-Net and what SEDA-Net does.

Communication Coordination in MAC. As explained in the previous sections, in the dual-radio setting of our interest, either/both of short- and long-range can be used as a radio that performs communication coordination, which we call "coordination radio". This coordination radio is the one which executes duty-cycling. As an asynchronous and sender-initiated duty-cycling, we use X-MAC [9] in our design. However, the choice of coordination mechanism is orthogonal in our study, and thus other methods can be applied to SEDA-Net. In this approach, TX first transmits a long (strobed) preamble signal using the coordination radio and RX periodically listens to the preamble and replies ACK to notify TX of RX's coordination success. Once the communication coordination is finished, actual data delivery occurs. Based on this basic X-MAC operation, we extend SEDA-MAC, which is a MAC layer subcomponent of SEDA-Net, for efficient communication coordination to the self-configuring protocol.

Routing for Lifetime Maximization. As a routing protocol that determines the next-hop of a node, we develop a new protocol that is a variant of RPL [18]. Different from the MAC protocols that are typically tailored to a target application, RPL is a *de facto* standard that is highly generalized by enabling us to define a customized objective function for a desired link metric. In our generalized dual-radio setup, RPL should be modified so as to determine which radio will be used for data transmission to the next-hop, which we call *data radio selection*. What is novel in SEDA-Net is to define a new OF for prolonging the network lifetime and include a mechanism to determine the data radio (i.e., main radio) for each link, for which we propose a new protocol SEDA-Routing, a routing-layer subcomponent of SEDA-Net.

SEDA-Net: What Needs to be Configured? Major issues in the self-configuring protocol include: (i) with which radio we perform the communication coordination (in MAC), and (ii) to which neighbor a TX needs to forward the data and with which radio (in routing). To discuss how these decisions are made more clearly, we first introduce some notations. We let V be the entire set of nodes in the network, and

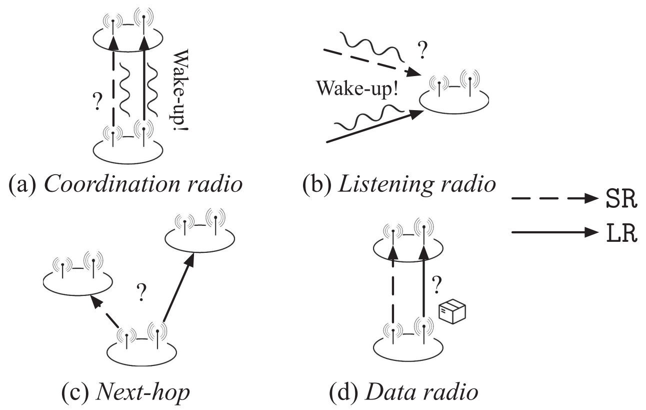


Fig. 2. SEDA-Net determines (a) which radio is used for sending an wake-up call, (b) which radio is used to receive an wake-up call, (c) next-hop reached by which radio, and (d) which radio is used for sending data.

use the symbols LR, SR, and LSR, to refer to the long-range, short-range, and both radios, respectively, and use NONE for no assignment. Then, as shown in Fig. 2, for each node $v \in V$, it is necessary to decide the following (from MAC and routing layers), which we call *(communication) configuration* throughout this paper:

a) *Coordination radio matrix* $[C_{vw}]_{v,w \in V}$, where $C_{vw} \in \{\texttt{LR}, \texttt{SR}, \texttt{NONE}\}$ is the radio choice for the communication coordination for the data transmission from v to w.
b) *Listening radio vector* $[L_v]_{v \in V}$, where $L_v \in \{\texttt{LR}, \texttt{SR}, \texttt{LSR}, \texttt{NONE}\}$ is the radio choice for periodically listening to some nodes' coordination request for possible data transmissions to v.
c) *Next-hop vector* $[N_v]_{v \in V}$, where N_v is the next-hop neighbor of v towards the base station.
d) *Data radio matrix* $[D_{vw}]_{v,w \in V}$, where $D_{vw} \in \{\texttt{LR}, \texttt{SR}, \texttt{NONE}\}$ is the radio choice for actual data transmission from v to w.

This perspective of opening all the possible combinations as in the above enables us to have a wide search space for a good solution. For example, for two nodes a and b with $N_a = b$, the configuration of $C_{ab} = \texttt{LR}$, $L_b = \texttt{LR}$, $D_{ab} = \texttt{SR}$ denotes the duty-cycled wake-up radio protocol which means the following: (i) the next-hop of a is b, (ii) a uses LR to send the coordination request (i.e., wake-up call) to b, (iii) b listens to LR to coordinate communication, and (iv) the actual data delivery from a to b is made over SR. The homogeneous radio configuration (i.e., only short- or long-range radio) can be represented by the configuration of $C_{ab} = \texttt{SR}, L_b = \texttt{SR}$, $D_{ab} = \texttt{SR}$ (resp. $C_{ab} = \texttt{LR}, L_b = \texttt{LR}$, $D_{ab} = \texttt{LR}$). One may ask what $L_v = \texttt{LSR}$ means. This implies that v listens to both short- and long-range radios to coordinate communication.

3.2 Goal

We let a configuration be $\mathcal{T}_{\text{cf}} = \{[C_{vw}], [L_v], [N_v], [D_{vw}]\}$, which is defined for a multigraph $G(V, \bar{E})$ due to two possible short- and long-range links between any two nodes. $E_{\text{init}}(v)$ denotes the initial energy of node v and the base station v_0 is assumed to have the infinite amount of energy. Also, let $E_\Delta(v) = E_\Delta(v, \mathcal{T}_{\text{cf}})$ be the consumed energy per unit time for node v, consisting of

$$E_\Delta(v) = E_{\text{TX\_Data}}(v) + E_{\text{RX\_Data}}(v), \tag{1}$$

where $E_{\text{TX\_Data}}(v)$ and $E_{\text{RX\_}\mathit{Data}}(v)$ correspond to the amount of energy consumptions for transmission and reception at node v per unit time, respectively. Defining the network lifetime as the time until we have the first node that runs out of its battery, we consider the following optimization problem, called **D-LTMAX**: for a given degree constant $m > 0$,

$$\begin{aligned} \mathbf{D-LTMAX:} \quad & \max_{\mathcal{T}_{\text{cf}}} \min_{v \in V} LT(v, \mathcal{T}_{\text{cf}}) \\ & \text{subject to } \texttt{Deg}(v) \le m, \quad \forall v \in V, \end{aligned} \tag{2}$$

where $LT(v, \mathcal{T}_{\text{cf}}) = \frac{E_{\text{init}}(v)}{E_\Delta(v, \mathcal{T}_{\text{cf}})}$.

The constant m is a parameter that controls how much contention is allowed around one node. As RPL constructs a tree (thus, a single parent for every node in the network), an internal node (except for the base station) is permitted to have one parent and at most $m-1$ children. Under carrier sensing mechanisms (e.g., CSMA) in wireless environments, the wireless medium utilization will be dramatically degraded as the number of contending nodes increases. Thus, the parameter m needs to be chosen with respect to the interference range and data rate of a node. We design SEDA-Net to be an approximating heuristic that tries to solve **D-LTMAX** in a distributed manner (see Section 4.5 for the rationale of SEDA-Net and NP-completeness of **D-LTMAX**).

Remark on Lifetime. Network lifetime in this paper is the time until there exists a node whose battery is exhausted. In the literature [42], [43], the network lifetime can be differently defined as: (i) until n nodes' batteries are exhausted, (ii) network is disconnected, or (iii) nodes' coverage becomes shrunken. The reason why we choose the definition of the network lifetime based on the first energy-depleted node is (a) this definition is the most popular in the literature due to its simplicity and tractability, and (b) the lifetime during which a network properly functions could be determined by a couple of nodes closely located to the base station in lots of data collection networks for IoT. Moreover, in Section 5.2, we evaluate a different definition of network lifetime as 20 percent-Lifetime until 20 percent of nodes in the network deplete their energy, which demonstrates that by solving **D-LTMAX** the lifetime can be improved not only for the first node, but also for a portion of nodes in the network. Thus, we believe developing a protocol based on the first battery exhaustion is of some value.

4 SEDA-MAC, SEDA-ROUTING, AND CROSS-OPT

4.1 SEDA-Net: Overall Framework

We first describe the overall framework of SEDA-Net by explaining what each node v performs, divided into two phases: *Init* and *Recalibration* (see Algorithm 1 and Fig. 3).

Init Phase. After each node v is initially deployed, v joins the network by receiving a DIO message (which is RPL's control packet for network advertisement), where v initially configures the listening radio as LR. After joining the network, v transmits DIO messages periodically, following the standard RPL. Once v receives other DIO messages from its neighbor, say u, it adds u to its neighbor set $\mathcal{S}_v$ and chooses the coordination radio $C_{vu} = \texttt{LR}$. v chooses its next-hop N_v

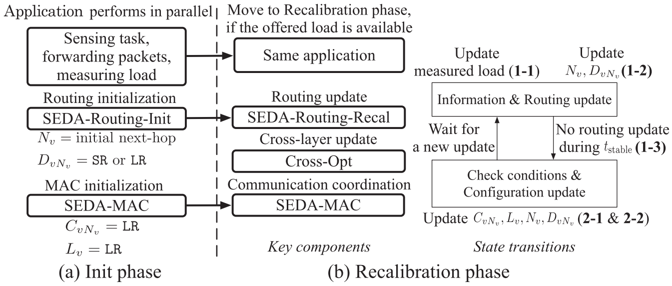


Fig. 3. The operation diagram of each node v in SEDA-Net. (a) describes how SEDA-Net is initialized in *Init* phase. (b) shows the key components and the state transitions of SEDA-Net in *Recalibration* phase.

and data radio D_{vN_v} using SEDA-Routing-Init that is a dual-radio extension of vanilla RPL [18] with OF0[20] (see Section 4.3). Once the initial N_v and D_{vN_v} are determined, v starts to perform sensing and generate data packets. While v forwards data packets, generated by itself or forwarded from its children, to N_v, v periodically measures the traffic load offered to itself. Here, the offered load denotes the amount of a traffic load on each node, measured by counting the number of packets passing through v and averaging it over a predefined time window t_m. When the first measurement of the offered load becomes available, it moves to *Recalibration phase* and switches the routing module from SEDA-Routing-Init to SEDA-Routing-Recal (see Section 4.3). Note that performing sensing task, forwarding data packets, and measuring the offered load continue in parallel with running SEDA-Routing. Whenever a data packet is transmitted, SEDA-MAC is used with the configuration of coordination, listening and data radios at that instant.

Recalibration Phase. Once initialization is finished, each node v runs SEDA-Routing-Recal, which is designed to further optimize the configuration, so as to prolong the network lifetime by balancing the traffic load better. While it runs SEDA-Routing-Recal, v embeds the offered load of v in DIO messages sent periodically. Thus upon receiving a DIO packet from u, it updates the link weight based on the offered load of u in the DIO packet (step **1-1**). Using the updated information, v finds the best next-hop N_v and data radio D_{vN_v} and waits for another DIO packet to keep updating (step **1-2**). If there is no change by SEDA-Routing-Recal during t_{stable} (step **1-3**), v runs the Cross-Opt protocol, which finalizes to find out an energy-efficient configuration from the cross-layer perspective (step **2**) (see Section 4.4). We choose t_{stable} from our empirical simulations or experiments (see Section 5). If there is no change in Cross-Opt, meaning that the configuration becomes stabilized, it waits until receiving the DIO with the updated information in step **1-1** of SEDA-Routing-Recal.

Self-Configuring and Distributed Operation. We comment that Init and Recalibration phases are just logically maintained inside a node, which does not require all nodes' phases in the network to be synchronized. The coordination and listening radios should be matched to coordinate communication and exchange packets. In Init phase, each node v chooses $L_v =$ LR and $C_{vu} =$ LR for all neighbors u. On the other hand, as a result of Cross-Opt, a link uw may have $C_{uw} =$ SR and $L_w =$ LSR. Since u uses SR for coordination only to w and w chooses both LR and SR for listening, any node can coordinate and exchange packets to any other neighbors, even though they are in different phases.

4.2 MAC Layer: SEDA-MAC

Coordination Solution Selection. The coordination and listening radios of each node are first initialized when the system starts, and then may be modified by Cross-Opt, which we separately describe in what follows. In both Init phase and Cross-Opt, we consider only two scenarios of configuration choices for a node: (i) only LR is used for both coordination and listening radios, and (ii) choosing LSR as listening radios, which we call *dual listening*, where the coordination radio can be chosen to be either of LR or SR. We note that the other options which choose SR as both coordination and listening radios are highly likely to require the multi-hop wake-up signalling to perform communication coordination with a far node. The multi-hop wake-up would add large complexity and/or energy waste on the protocol, therefore those options are ignored in our design.

Algorithm 1. What Each Node v Performs in ***SEDA-Net***

Init phase
  Once v joins the network, it runs **SEDA-Routing-Init** which outputs initial N_v and D_{vN_v} based on OF0.
  v sets $L_v =$ LR and $C_{vu} =$ LR for all u, neighbors of v.
  If the first measurement of offered load is available,
  it switches the routing module to **SEDA-Routing-Recal** and goes to **Recalibration phase**.
Recalibration phase
  1. SEDA-Routing-Recal
    1-1. *Routing information update:* Upon receiving a DIO message from u, v updates the measured load of u from the DIO.
    1-2. *Routing update:* Based on the updated load, v chooses N_v and D_{vN_v} using **SEDA-Routing-Recal** (see Section 4.3) and goes to step **1-1**.
    1-3. *Stabilization:* If there is no update in both N_v and D_{vN_v} in step **1-2** during t_{stable}, v goes to step **2.**
  2. Cross-Opt
    2-1. *Check conditions:* For every $u \in \mathcal{S}$ of v, v checks two conditions **C1** and **C2** that imply whether the network lifetime increases by applying **Cross-Opt** see Section 4.4).
    If v finds u that satisfies the conditions, goto step **2-2**, otherwise goes to step **1-1**.
    2-2. *Configuration update:* v updates its configuration by **Cross-Opt** and goes to step **1-1**.
Note. All communication coordinations for data forwarding and DIO transmission are performed by **SEDA-MAC** at Layer 2 using the configuration at each corresponding step.

(a) Init phase. When the system starts, we first use LR for both coordination and listening radios for all nodes. This is because (i) we need to have a match between the coordination and the listening radios for communication coordination, and (ii) at the initial stage, it seems to be a good choice for each node to have a wide reachability in coordination, and thus to be assigned an opportunity to search as many nodes as possible, as a next-hop candidate.

(b) Cross-Opt. Cross-Opt runs whenever the configuration for the routing module (i.e., data radio and next-hop) is stabilized. We design Cross-Opt to guarantee that a given TX-RX pair (from routing) has matched the coordination and the listening radios. The result of Cross-Opt often generates some set of nodes, each of which turns on *both long- and short-range radios as listening radios*. By doing that the loads are balanced more efficiently, so that it can prolong the network lifetime (see Section 4.4 for details).

Communication Coordination. Another key role of MAC is a way to actually coordinate a pair of TX and RX so that they are awake at the same time. As in most asynchronous MAC protocols (especially the sender-initiated ones), periodic listening and preamble are the main ingredients of communication coordination, for which we use that of X-MAC [9] as a baseline mechanism, but add more designs for the dual-radio setup. We describe our communication coordination for different types of traffic: unicast and broadcast.

(1) Unicast. The communication coordination for unicast traffic can be done using either of long- or short-range radios, depending on what our Cross-Opt produces. We simply use the mechanism in X-MAC, with the following addition: we add one more bit to the strobed preamble packet to represent the information on the data radio, where a TX node sets the bit as 0 if the data radio is SR; otherwise, the bit is set to 1.

(2) Broadcast. Communication coordination for broadcast traffic requires time-synchronization with every neighbor, for which (i) we remove the preamble ACK to ensure that every neighbor is set to be awake with the TX, (ii) the TX transmits the preamble up to the maximum preamble length, and (iii) we use only the long-range radio for coordination. We also have the following two design ideas for better energy efficiency.

- *End-of-preamble prediction.* In the broadcast preamble, we embed the sequence number for each strobe packet, and thus the maximum sequence number for the maximum preamble length is known to every node *a priori*. Using this, after receiving the broadcast preamble, the receiver can switch off its radio until the end of the preamble, thereby reducing the energy consumption of a long waiting time until all neighbors are coordinated.
- *Back-to-back transmissions in broadcasting routing msgs.* It is required to broadcast the routing control messages over both long- and short-range radio links so as to measure the qualities for all possible neighboring links and exchange the routing information for both SR and LR. We perform this broadcasting with a single preamble through back-to-back transmissions. To be more precise, the sender first transmits the broadcast preamble for long broadcast, and receivers get the preamble and wait until the broadcast packet transmission using end-of-preamble prediction. Since all neighbors having a short-range radio of the TX are synchronized, they turn on the short-range radio for incoming short broadcast, right after receiving the long broadcast.

Medium Access Control and Reliability. As in many other protocols in low-power networks, we choose a CSMA-CA based mechanism. TX listens to the medium to check out other transmission activities prior to and during the preamble transmission. If TX fails to receive the preamble ACK or the data ACK, it performs multiple retransmissions for a pre-specified number of maximum retransmission count or successful transmission with exponential backoffs. If it receives other preambles rather than preamble ACK, it immediately stops transmitting preamble packets and goes into the retransmission phase.

4.3 Routing Layer: SEDA-Routing

SEDA-Routing is based on RPL [18]. SEDA-Routing consists of two protocols: SEDA-Routing-Init and SEDA-Routing-Recal, where we apply a different weight to each of those two, to form an energy-efficient routing tree, considering the dual-radio setup.

SEDA-Routing-Init. We run SEDA-Routing-Init, when the sensors are first deployed, for which SEDA-Routing-Init is RPL with OF0, but with different weights $w_{vu(L)}$ and $w_{vu(S)}$ [2] for each link vu when established using long- and short-range radios, respectively. Different weights are due to the fact that the two radios have different characteristics in terms of power consumption, bandwidth, and communication range. We simply set $w_{vu(S)} = \text{ETX}_{vu}$ by ETX, but $w_{vu(L)} = k \times \text{ETX}_{vu}$, where the constant k is a tunable parameter depending on the specification of the long-range radio chip. Finally, to remove the herd behavior,[3] when the new next-hop is updated, it is chosen with probability $1/2$.

SEDA-Routing-Recal. When SEDA-Routing-Recal runs, we use different weights from those in SEDA-Routing-Init, which we denote by $\tilde{w}_{vu(L)}$ and $\tilde{w}_{vu(S)}$ to achieve longer lifetime. We design the weight $\tilde{w}_{vu(L)}$ (resp. $\tilde{w}_{vu(S)}$) as

$$\tilde{w}_{vu(L)} = w_{vu(L)} \times \Big(\texttt{glb\_load}(u) + \texttt{loc\_load}(u)\Big), \quad (3)$$

where

$$\begin{aligned} &\texttt{glb\_load}(u) = \alpha * \texttt{glb\_load}(u) + (1-\alpha) * \texttt{dcount}(u), \\ &\texttt{loc\_load}(u) = \beta * \texttt{Deg}(u). \end{aligned} \quad (4)$$

The new weight in (3) over a link vu additionally considers the offered load at node u, whose role is to avoid routing packets to the nodes that are heavily loaded. The load is measured by a combination of global and local loads. The role of global load $\texttt{glb\_load}(u)$ quantifies how many packets u has transmitted and received, which we measure by an exponential moving averaged $\texttt{dcount}(u)$ (i.e., the number of data packets passing through u) with moving average weight α. The local load $\texttt{loc\_load}(u)$ is used to avoid the node suffering from heavy contentions and interferences, which we simply measure by node u's degree $\texttt{Deg}(u)$. For Rank, we also use the hop count as in SEDA-Routing-Init, and the next-hop of a node v is updated to be the one that has the minimum $\texttt{Rank}(u) + \tilde{w}_{vu(L)}(\text{resp. } \tilde{w}_{vu(S)})$. In addition to the parameter k in the weight, β is also the parameter which controls the impact of glb_load and loc_load on the

2. We use the notations $vu(L)$ and $vu(S)$ to refer to LR and SR link vu, respectively.
3. A set of nodes changes its next-hop to the same node simultaneously, so that they experience severe congestion.

routing. We will investigate the impacts of those parameters and the way of choosing proper values through our evaluation (see Section 5.2).

Again, to avoid the herd effect, we employ a probabilistic choice of the next node, but with the following rule to consider load balancing. Out of two choices, the current next-hop N_v and the new next-hop u (if it differs from N_v), the probability p_{vu} that v chooses u as a next-hop is set to be $(1 - \frac{1}{\texttt{Load\_diff}})$, where Load_diff measures the global and local load differences due to the change of the next-hop to u, calculated by

$$\begin{aligned}\texttt{Load\_diff} = |(&\texttt{glb\_load}(N_v) + \texttt{loc\_load}(N_v)) \\ &- (\texttt{glb\_load}(u) + \texttt{loc\_load}(u))|.\end{aligned}$$

Thus, node v changes the next-hop with high probability when the current and the new next-hop have the large load difference.

When each node v updates the next-hop u according to the weight $\tilde{w}_{vu(L)}$ and $\tilde{w}_{vu(S)}$, the data radio D_{vu} needs to be chosen to minimize energy consumption. If the next-hop u is reachable by both short- and long-range radios, D_{vu} is chosen as SR when the energy consumption for data transmission by the short-radio is smaller than that by the long-range radio. Otherwise, the data radio D_{vu} should be LR, including the case where the next-hop u is only reachable by the long-range radio. By doing so, we achieve not only load balancing by the next-hop update, but also energy minimization by the data radio selection.

While all sensors perform the sensing task and forward data packets, each node v updates glb_load(v) and loc_load(v) for every data collection. We embed four more information in a DIO message of a node, say v, in addition to those specified by the standard RPL, glb_load(v), loc_load(v), $E_{\text{rem}}(v)$, and $E_{\Delta}(v)$. The values of $E_{\text{rem}}(v)$, and $E_{\Delta}(v)$ are the remaining energy and the energy consumption rate of v, respectively. For every reception of DIO messages, each node updates the configuration as finding the best next-hop with the data radio.

Robustness. To ensure robustness, SEDA-Routing inherits RPL's repair module [18]. When loops are detected, SEDA-Routing runs local or global repair modules to remove loops. Due to abrupt network changes (e.g., mobility, link failure, etc.), if each node immediately needs to update routing, by increasing the frequency of DIO exchange, the node can speed up the response to those changes. Furthermore, by rejecting bad links which have ETX worse than a certain threshold, SEDA-Routing tries to make a reliable and robust routing tree, if there exist better links.

4.4 Cross Layer: Cross-Opt

In this section, we propose Cross-Opt, which further optimizes the *configuration* involving both MAC and routing layers. To illustrate the motivation of Cross-Opt, we consider the following example, as illustrated in Fig. 4.

Example. Assume that in both cases (a) and (b) of Fig. 4, nodes v and u can communicate using SR. In (a), BS (denoted by s) typically is equipped with a power supply, and thus do not need to perform duty-cycling, so that u is able to transmit data packets without any coordination. This leads u to consume much less energy than v. In (b), v

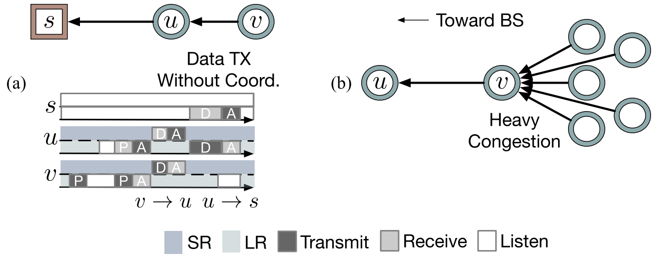


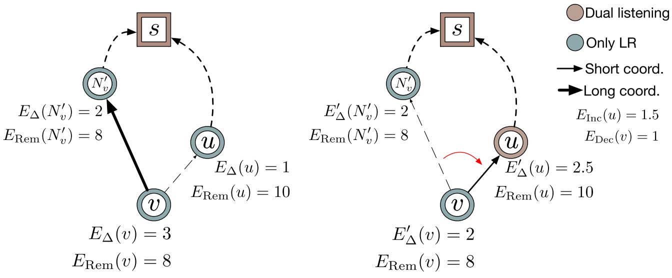
Fig. 4. Two examples: How Cross-Opt helps in prolonging network lifetime. In both examples, v consumes more energy than its next-hop u due to (a) u is directly connected to BS (denoted by s) or (b) v suffers from heavy congestion; thus applying Cross-Opt can improve network lifetime.

has many children, thereby rendering v waste much energy due to the control of contentions among its children's transmission attempts. In these cases, we consider the option of making u turn on its long- and short-range radios together (i.e., *dual listening*), so that we have $L_u = \texttt{LSR}$ and $C_{vu} = \texttt{SR}$. This has an effect of more energy consumption at u (due to dual listening) but less energy consumption at v (due to short-range radio based coordination). It is sometimes helpful in balancing the consumed energy in the network, and thus efficient in prolonging the network lifetime if the prolonged lifetime of v exceeds the original lifetime of both v and u.

How Cross-Opt Works. We now elaborate on the conditions under which Cross-Opt changes the configuration in a distributed manner, as illustrated in the aforementioned example. We first let $S(v)$ be the set of neighbor nodes of a node v connected by SR. Then, each node v checks the following conditions **C1** and **C2** for each $u \in S(v)$

$$\begin{aligned}\textbf{C1}: \quad & \frac{E_{\text{rem}}(u)}{E_{\Delta}(u) + E_{\text{Inc}}(u)} > \min_{i \in (v,u,N_v)} \left[\frac{E_{\text{rem}}(i)}{E_{\Delta}(i)}\right], \\ \textbf{C2}: \quad & \frac{E_{\text{rem}}(v)}{E_{\Delta}(v) - E_{\text{Dec}}(v)} > \min_{i \in (v,u,N_v)} \left[\frac{E_{\text{rem}}(i)}{E_{\Delta}(i)}\right]\end{aligned}$$

where E_{rem} and E_{Δ} are the current remaining energy embedded and exchanged in DIO messages, as mentioned earlier. We denote by $E_{\text{Inc}}(u)$ the additional amount of increased energy consumption rate, if u performs dual listening. Then, the condition **C1** implies that the decreased lifetime of u due to u's dual listening exceeds the minimum of original lifetimes of nodes $\{v, u, N_v\}$. Similarly, the condition **C2**means that the increased lifetime of v also exceeds the original lifetime of nodes, where $E_{\text{Dec}}(v)$ corresponds to the reduced energy consumption rate by changing its long-range radio coordination to the short-range radio one. $E_{\text{Inc}}(u)$ and $E_{\text{Dec}}(v)$ can be obtained from measurements. If **C1**and **C2** are satisfied for node u, node v updates its configuration as: (i) for coordination radio over link vu, set $C_{vu} = \texttt{SR}$, (ii) for listening radio of u, set $L_u = \texttt{LSR}$ (for which u is notified via a control packet from v), (iii) for the next-hop of v, set $N_v = u$ if $N_v \neq u$, and (iv) for data radio over link vu, set $D_{vu} = \texttt{SR}$ if $w_{vu(S)} \leq w_{vu(L)}$, otherwise $D_{vu} = \texttt{LR}$. If there are multiple u satisfying the above conditions, v chooses the node that makes the longest lifetime of the network. Fig. 5 exemplifies the effect of Cross-Opt, where the lifetime increases from 2.667 to 4.



(a) Without Cross-Opt (b) Applying Cross-Opt

Fig. 5. Prior to applying Cross-Opt, the network lifetime (determined by v's lifetime) is 2.667 (=8/3). Satisfying **C1** and **C2** for node u, v changes its configuration following Cross-Opt, resulting in the lifetime of 4 (=8/2).

4.5 SEDA-Net and D-LTMAX

We now present the rationale of SEDA-Net, for which recall our goal of **D-LTMAX** in Section 3.2 and the consumed energy per unit time for node v is given as: $E_{\Delta}(v) = E_{\text{TX\_Data}}(v) + E_{\text{RX\_Data}}(v)$. Then, considering how routing and MAC operate, we have

$$E_{\text{TX\_Data}}(v) = (\texttt{dcount}(v) + 1) * (P_{\text{Coord}} + P_{\text{TX\_Data}}),$$
$$E_{\text{RX\_Data}}(v) = \texttt{dcount}(v) * (P_{\text{Listen}} + P_{\text{RX\_Data}}),$$

where P_{Coord} and P_{Listen} correspond to the consumed energies for coordination, and $P_{\text{TX\_Data}}$,$P_{\text{RX\_Data}}$ are the ones for data transmission/reception of one packet, and the number of packets traversing v per unit time is $\texttt{dcount}(v)$. Then, Theorem 4.1 states that **D-LTMAX** is hard to solve even with a centralized method.

Theorem 4.1. *Given a multigraph $G(V, \bar{E})$, where each node $v \in V$ is equipped with both long- and short-range radios, each node v generates a data packet per unit time and collects those packets to the base station v_0 without data aggregation.*[4] *Then, D-LTMAX problem is NP-Complete.*

The key step in the proof of Theorem 4.1 lies in the reduction from the problem of capacitated minimum spanning tree [44], which is to find a minimum spanning tree such that the size of every subtree directly connected to the root is smaller than some $K > 2$. We comment that a similar lifetime maximization problem has been addressed in [24], [25], [26], but **D-LTMAX** has the following differences: (i) extension to a dual-radio setup by modeling it as a multigraph $G(V, \bar{E})$ and (ii) consideration of the contention degree through the constant m.

Proof of Theorem 4.1. For simplicity, we consider a simpler case of $E_{\text{init}} = E_{\text{init}}(v)$ for all $v \in V$, which suffices to show the result. Then, **D-LTMAX** can be rewritten as follows:

$$\min_{\mathcal{T}_{\text{cf}}} \max_{v \in V} \frac{1}{E_{\text{init}}} \times E_{\Delta}(v) \quad \text{subject to} \quad \texttt{Deg}(v) \le m, \quad \forall v \in V. \tag{5}$$

The recasted **D-LTMAX** into a decision problem is to determine whether there exists configuration $\mathcal{T}_{\text{cf}}$ satisfying $\max_{v \in V} \frac{1}{E_{\text{init}}} * E_{\Delta}(v) \le K'$ with the degree constraint m, where K' is a given constant.

4. Data aggregation means the ability to aggregate multiple data packets into a single packet.

To prove Theorem 4.1, we use reduction from the capacitated minimum spanning tree to the **D-LTMAX** decision problem. We start with constructing an instance of **D-LTMAX** from an instance of the capacitated minimum spanning tree as follows. Given the graph $G(V, E)$, we construct G' with the same V and $\bar{E}$ such that every $e_{vw(S)} \in E$ is duplicated, satisfying $e_{vw(S)} \in \bar{E}$ and $e_{vw(L)} \in \bar{E}$. Let $E_{\text{init}} = 1$, $P_{\text{Coord}} + P_{\text{TX\_Data}} = 1$ and $P_{\text{Listen}} + P_{\text{RX\_Data}} = 1$. This construction can be done in polynomial time. We now show that G has a minimum spanning tree with capacity $\le K$, if and only if G' has configuration such that $E_{\Delta}(v)$ for all v is smaller than or equal to $2K - 1$ with $\texttt{Deg}(v) \le K$.

($\longrightarrow$): To build a non-trivial configuration, let the degree of the base station v_0 be bounded by K, so that it can avoid severe congestion at v_0. Suppose that G has a capacitated minimum spanning tree T. We construct a configuration such that, for all $e_{vw} \in T$, $N_v = w$ and D_{vw}, C_{vw} and L_v are $\texttt{LR}$ (or $\texttt{SR}$). Since the size of subtree is bounded by K, $\texttt{Deg}(v)$ is also bounded by K for any $v \in V \setminus v_0$ and $\texttt{Deg}(v_0) \le K$ by the assumption. Since it is assumed that every node in the graph generates a data packet per unit time, we have $\texttt{dcount}(v) = \|\text{subtree}(v)\| - 1 \le K - 1$ by definition, where $\|\text{subtree}(v)\|$ is the size of subtree rooted by v and $E_{\Delta}(v) = 2 * \texttt{dcount}(v) + 1 \le 2K - 1$ for all v with $\texttt{Deg}(v) \le K$.

($\longleftarrow$): Suppose that G' has configuration that satisfies $E_{\Delta}(v) \le 2K - 1$ with $\texttt{Deg}(v) \le K$, then we can construct a minimum spanning tree T based on N. Then, for any v, since $E_{\Delta}(v) = 2 * \texttt{dcount}(v) + 1 \le 2K - 1$, $\|\text{subtree}(v)\| \le K$.

This completes the proof.

SEDA-Net: A Distributed Heuristic to Solve D-LTMAX. The goal of D-LTMAX is to minimize $E_{\Delta}(v)$ with the degree constraint m under the assumption of $E_{\text{init}}(v) = E_{\text{init}}$, $\forall v \in V$. By moving the degree constraint to the objective function using $\bar{\beta}$ as its price, we have a new objective function $E_{\Delta}(v) = E_{\text{TX\_Data}}(v) + E_{\text{RX\_Data}}(v) + \bar{\beta} * \texttt{Deg}(v)$. In SEDA-Routing-Recal, each node v tries to minimize not only the weight on the link $w_{vu(L)}$ (or $w_{vu(S)}$), but also $\texttt{glb\_load}(u) + \texttt{loc\_load}(u)$ in a distributed manner by choosing the next-hop u that has minimum $\tilde{w}_{vu(L)} = w_{vu(L)} \times (\texttt{glb\_load}(u) + \texttt{loc\_load}(u))$ (or $\tilde{w}_{vu(S)}$) (see Section 4.3). The effort of SEDA-Routing-Recal to minimize $\texttt{glb\_load}(v)$ aims to minimize $E_{\text{TX\_Data}}(v) + E_{\text{RX\_Data}}(v)$ of D-LTMAX at which our SEDA-Net tries to find the configuration $\mathcal{T}_{\text{cf}}$ such that the number of descendants is minimized. Minimizing $\texttt{loc\_load}(v)$ defined by $\beta * \texttt{Deg}(v)$ attempts to satisfy the degree constraint or minimize $\bar{\beta} * \texttt{Deg}(v)$, which implies that the degree of the given configuration $\mathcal{T}_{\text{cf}}$ is bounded properly with the tunable parameter β.

SEDA-Net: Complexity Analysis. In order to maximize the network lifetime, SEDA-Net requires $\texttt{glb\_load}$ and $\texttt{loc\_load}$ which are embedded in a DIO message. First of all, while data packets are forwarded, $\texttt{glb\_load}$ is measured accordingly. Depending on traffic patterns of IoT applications (e.g., periodic, bursty, or event-driven), the required time for $\texttt{glb\_load}$ measurement can be highly different. Once $\texttt{glb\_load}$ becomes available, the message complexity of SEDA-Net is determined by how many DIO messages are exchanged until SEDA-Net converges.

TABLE 1
Specification of CC1200 and CC2538

	CC1200	CC2538
Frequency band	920 MHz	2.4 GHz
Data rate	50 kbps	250 kbps
TX power	108 mW	72 mW
RX power	69 mW	60 mW
RX sensitivity	-109 dBm	-97 dBm

When the number of nodes in the network is n with the maximum hops h (from a node to BS), the complexity until glb_load and loc_load are synchronized requires $\mathcal{O}(h)$. Given those loads, we apply the probabilistic choice of the next-hop with the probability $(1 - \frac{1}{\texttt{Load\_diff}})$, and thus the network converges when Load_diff for all nodes becomes smaller than a threshold (heuristically determined). In the worst case, Load_diff decreases by 1 per iteration – the load synchronization followed by the change of the next-hop. Thus, the total message complexity will be $\mathcal{O}(nh)$. It is worth noting that the order of the other static protocols' message complexity is the same as $\mathcal{O}(nh)$; however, the practical complexity depends on the radio configuration due to the transmitted range of DIO messages and the number of next-hop candidates.

In terms of the meassage overhead, the amount of bytes embedded in the DIO message for SEDA-Net is 4 bytes; thus, the overhead caused by SEDA-Net is minimized. We evaluate the complexity in Section 5.2 by presenting the convergence time. In brief, although SEDA-Net requires more time for the convergence (see Fig. 9), we believe it is worth to do for lifetime improvement.

5 Implementation and Evaluation

5.1 Setup

Implementation. We implement SEDA-Net as a set of new network drivers for Contiki OS [19], which is an open-source operating system for IoT devices and enables us to carry out both simulations and real experiments with the same implementation codes. The Contiki OS is originally designed only for a single radio setup, thus we extend the network drivers to support dual radios by modifying the radio interfaces. SEDA-MAC is implemented at Radio Duty Cycle and CSMA layers inside Contiki OS, taking charge of duty-cycling and medium access control. The neighbor management and RPL-related functions are also extended to support dual radios, including a new objective function (i.e., LTMAX) of our SEDA-Routing. The full source code is available in [45].

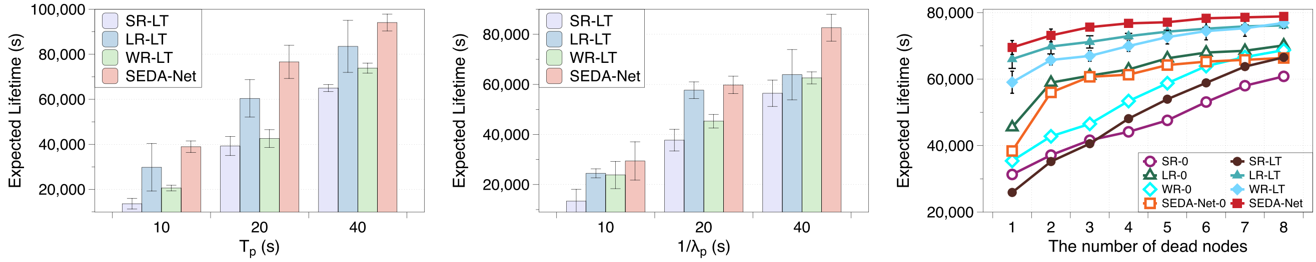
For real implementation, we use Firefly motes [3], having built-in dual-radio chips: CC2538 and CC1200 for SR and LR, respectively, where the characteristics of CC2538 and CC1200 are summarized in Table 1. To perform real experiments, we build a testbed with 45 Firefly motes [3] in a five-story office building as depicted in Fig. 13. Each mote in the testbed monitors all activities of the MCU and the radios to measure the energy consumption using the energy estimator module in the Contiki OS. We also perform extensive simulations using the Cooja simulator in the Contiki OS to test various scenarios under an controlled environment.

Traffic, Topology, and Metric. We consider two traffic models, periodic with period T_p and Poisson with mean rate λ_p, which can be regarded as periodic monitoring and event-driven application scenarios, respectively. We choose the data packet size of 50 bytes. In the simulations, we use GRID and RANDOM topologies, as in Figs. 7a and 7b, where GRID is with 36 nodes spaced by 50 m and RANDOM is with 50 nodes in 700 m x 700 m deployed uniformly at random. We choose GRID to test SEDA-Net in a regular and controlled case (e.g., the degree of each node is naturally bounded), while RANDOM is to evaluate the performance of SEDA-Net under more practical situations. We base our simulation on Multi-path Ray tracing Model (MRM) which models radio hardware properties (e.g., transmission power, receiver sensitivity), background noise, and multi-path interference (by obstacles) through SINR. Our primary performance metric is the network lifetime for which we examine the first energy depletion of one node as well as 20 percent of the entire nodes. Since the radio's duty cycle (i.e., the fraction of time when the radio is turned on) is the key to energy efficiency, the duty cycle for all tested protocols is optimized for the minimum energy consumption (i.e., as low as possible) while it should be larger than a certain threshold (given by an application) to guarantee data packet delivery. We also plot delay and PRR (Packet Reception Ratio), measured by the average time between packet generation and its reception at BS and the ratio of the number of received packets at BS, respectively.

Tested Protocols. We mainly compare SEDA-Net with state-of-the-art non-adaptive protocols based on a single radio (short- and long-range only, as SR and LR) and dual-radio (wake-up radio [13], [14], as). SR (resp. LR) configures a network only with short-range radios (resp. long-range radios). In other words, SR (resp. LR) chooses SR (resp. LR) as all for coordination, listening, and data radios. In terms of WR, we choose coordination and listening radios as LR and data radio as SR, which corresponds to a version of implementation of DCW-MAC [13], [14], fine-tuned to our environment. We note that to the best of our knowledge, there is no prior work on adaptively and heterogeneously configuring the dual-radio network. Thus, we choose DCW-MAC as the static and homogeneous configuration of dual radios to validate the advantages of SEDA-Net with heterogeneous and adaptive configuration. We also test two OFs of RPL; OF0 and LTMAX of SEDA-Net. To summarize, we test eight protocols: SEDA-Net, SEDA-Net-0 (i.e., SEDA-Net with only SEDA-Routing-Init), SR-LTMAX, SR-0, LR-LTMAX, LR-0, WR-LTMAX and WR-0, where in the protocol name 'X-Y', X denotes the radio configuration and Y represents the objective function. While SEDA-Net adopts a newly designed SEDA-MAC for MAC, the single radio protocols choose CXMAC, which is an X-MAC [9] based protocol in the Contiki OS. In the real testbed, to evaluate the wake-up radio protocol, we use CC1200 as the duty-cycled wake-up radio and CC2538 as the data radio.

5.2 Results: Simulation

Network Lifetime. Figs. 6a and 6b show the network lifetime at GRID and RANDOM with varying T_p and λ_p, respectively, where, for simplicity, LTMAX is denoted as LT. We test all



(a) Lifetime: Periodic and GRID (b) Lifetime: Poisson and RANDOM (c) 20%-Lifetime: Periodic with $T_p = 15$ and GRID

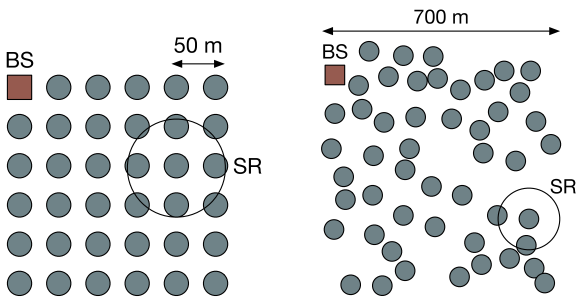
Fig. 6. Simulation-based lifetime evaluation of SEDA-Net. The black error bars denote the standard deviation over 10 random seeds.

LTMAX protocols with varying T_p and $1/\lambda_p$ as 10, 20, and 40 sec. Under low and high traffic densities for both periodic and Poisson, SEDA-Net shows at most 186 percent lifetime increase (i.e., 13,615 seconds of SR-LT is improved to 38,952 seconds of SEDA-Net for $T_p = 10$), which comes from choosing the proper configuration adaptively for given environment compared to non-adaptive solutions. Fig. 6c shows the performance of the 20 percent-lifetime (i.e., the time until 8th node dies) in GRIDunder $T_p = 15$. We observe that in almost all cases, LTMAX exhibits the longest lifetime than OF0, because, by balancing traffic loads just in non-adaptive solutions, LTMAX helps a lot with increasing the lifetime as well as balancing energy consumption among nodes. SEDA-Net achieves not only longer lifetime but also well-balanced energy consumption by solving **D-LTMAX**. On the other hand, other solutions (especially SR) have large gaps between the time until the first and the eighth dead node, thereby implying a large imbalance in the energy consumption among the nodes. That is, the balanced energy consumption by SEDA-Net leads to the improved lifetime of the 20 percent nodes as well as the first node.

Delay and PRR. Fig. 8 shows PRR and delay of all LTMAX protocols (SEDA denotes SEDA-Net) with periodic high and low traffic intensities (i.e., $T_p = 10$ and 20) in RANDOM, where bars represent PRR and lines denote delay. While the averaged delay decreases with SEDA-Net, the reliability measured in PRR becomes improved up to 99.348 percent. Those improvements are achieved by choosing the best configuration among both long- and short-range links while avoiding severe contentions and interferences.

Complexity and Scalability. Fig. 9 shows the time until the routing converges after initialization of all protocols with LTMAX in both GRID and RANDOM. To measure the practical convergence time, our measurement excludes trivial oscillations, which have little impact on the performance (e.g., only few nodes, located at leaves in the tree, keep switching paths due to minor differences). Due to the large searching space compared to SR configurations, SEDA-Net takes more time to converge (i.e., 1086 and 941 seconds, respectively). However, as we show that by running SEDA-Net, the network lifetime can be improved at most 186 percent compared to other non-adaptive configurations. Furthermore, since our protocol can deliver generated packets along slightly less efficient paths during the self-configuring duration, the convergence duration also can be regarded as the network running time. Thus, under stationary environments, we believe it is worth running SEDA-Net for less than 1 hour to build more energy-efficient networks and scalable for different size of topologies, evidenced by 36 and 50 nodes in the topologies with the similar convergence time.

Adaptive Self-Configuration. Fig. 10 depicts how SEDA-Net adaptively produces the self-configuring network configurations depending on given environments. In Fig. 10, (a) shows normal environment in which we perform the



(a) GRID: 36 nodes (b) RANDOM: 50 nodes

Fig. 7. Simulation topologies: GRID and RANDOM.

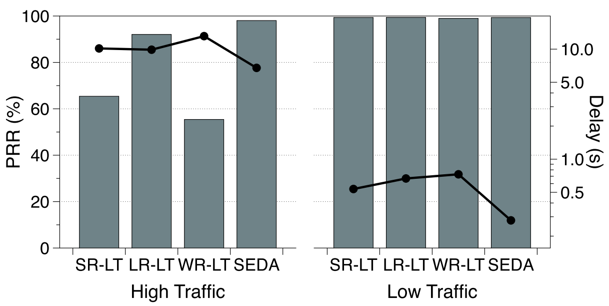


Fig. 8. PRR and Delay evaluation in Periodic with RANDOM.

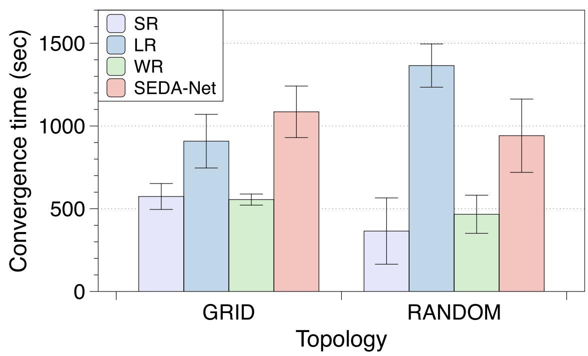
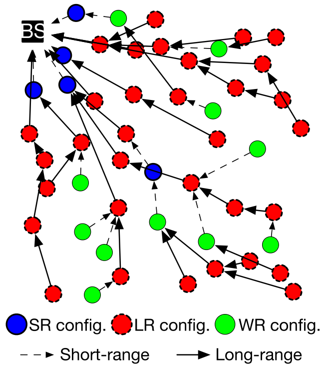
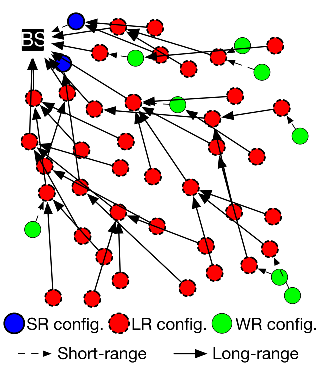


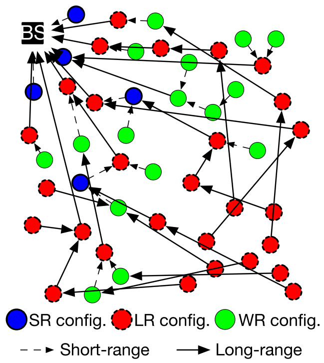
Fig. 9. Convergence time in GRID and RANDOM.



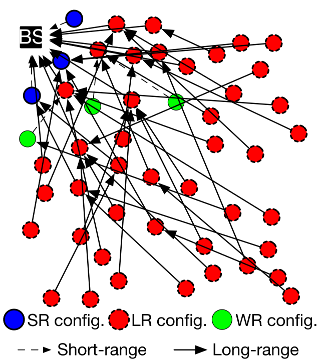
(a) Normal environment



(b) Wi-Fi interference



(c) Low-power LR



(d) High-power LR

Fig. 10. Self configuration under different cases (e.g., external interferences, different TX powers of LR). Different colors denote different configurations (SR, LR, and WR, respectively), while short or long-range radio is represented by dotted or solid lines, respectively.

simulations and (b) is the case of severe Wi-Fi interferences by increasing the background noise at 2.4 GHz frequency. We observe that SEDA-Net leverages more long-range radio links than short-range radio ones, and SR configuration is used only when the nodes' distance is close enough to overcome the external interference. (c) Low-power LR and (d) High-power LR correspond to low and high transmission powers and RX sensitivities of the long-range radio with 3 dBm differences. As shown in Figs. 10c and 10d, depending on the different RF characteristics, SEDA-Net properly chooses the network configurations composed of the different amount of long-range radio links.

Load Balancing. Fig. 11 shows the load balancing performance measured by the maximum number of descendants

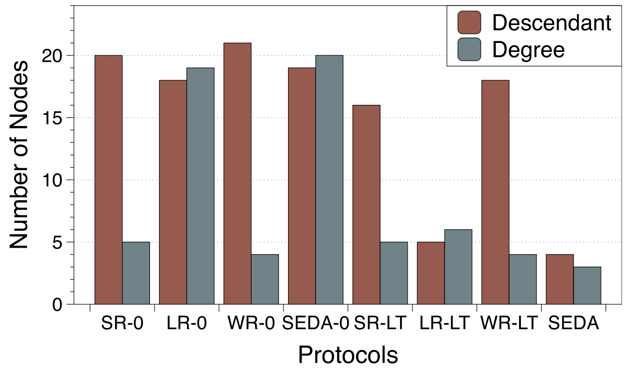
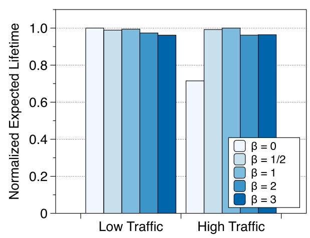
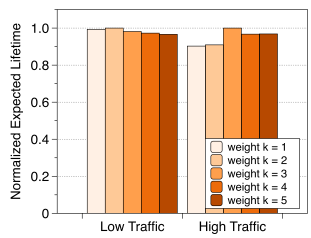


Fig. 11. The maximum number of descendants and degree in routing topology: RANDOM.



(a) Parameter Selection for β (GRID)

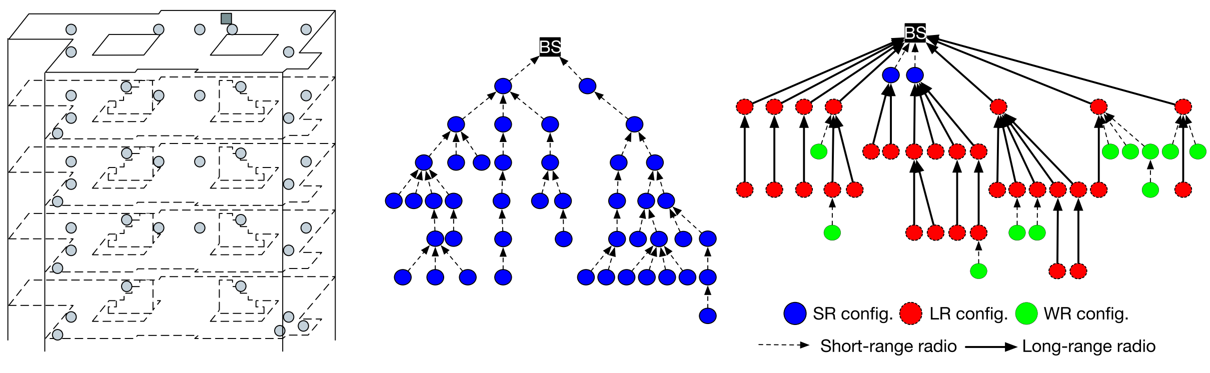


(b) Parameter Selection for k (GRID)

Fig. 12. Microbenchmark for parameter selection.

and the maximum degree (except the base station) in RANDOM. While all protocols with OF0 have a larger maximum number of descendants, choosing LTMAX can significantly reduce both the number of descendants and the degree. We note that SEDA-Net achieves the minimum size of descendants and degree.

Impact of β and k. Fig. 12 shows the impact of parameters β of loc_load and k of long-range radio weight $w_{vu(L)}$ and $\tilde{w}_{vu(L)}$ to the lifetime in SEDA-Routing under both low and high periodic traffics (e.g., $T_p = 60$ and 5, respectively). Fig. 12a shows the normalized lifetime of SEDA-Net varying β given weight $k = 2$. Because for low traffic the amount of contentions and interferences are negligible, the smaller β is chosen, the longer the lifetime is shown. It is obvious that for high traffic $\beta = 0$ which does not consider contentions shows the worst performance, so that choosing $\beta = 1$ can be a proper choice. Similarly, the normalized lifetime for varying weight parameter k given $\beta = 1$ is shown in Fig. 12b.



Sensor map in a five-floor bldg. ■=BS Routing Topology of SR-0 Routing Topology of SEDA-Net

Fig. 13. Sensor map and routing topologies in our testbed.

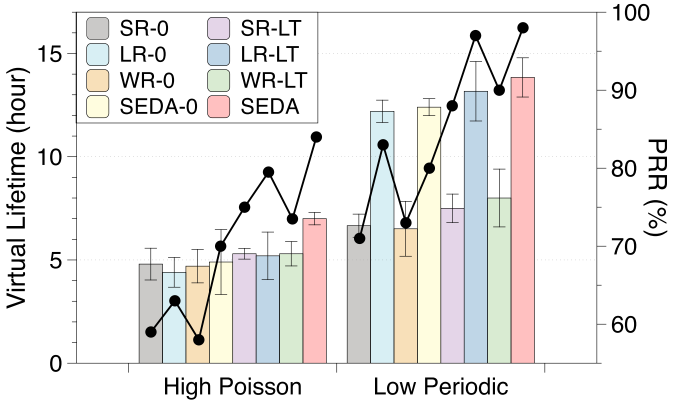


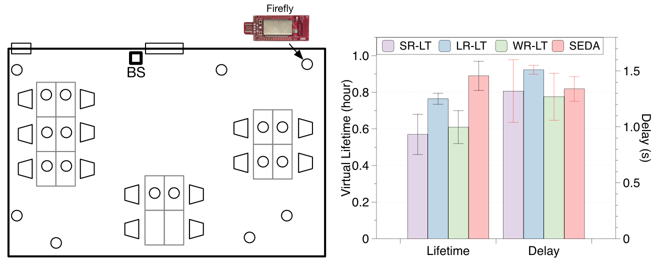
Fig. 14. Real testbed evaluation where virtual lifetime and PRR are denoted in bars and lines, respectively.

Since the small weight forces SEDA-Routing to prefer the long-range radio compared to the large weight (e.g., short- and long-range radios have the same weight if $k = 1$), under low traffic choosing the small weight (e.g., 1 or 2) shows better performance due to negligible contentions of long-range radios. However under high traffic the number of long-range links should be restricted, so that choosing the larger weight ($k = 3$ in our case) shows the longer lifetime.

5.3 Results: Real Experiment

Static Case. To evaluate all eight protocols under both high and low traffic patterns, we set $\frac{1}{\lambda_p} = 15$ and $T_p = 60$ for event-driven and periodic monitoring applications, denoted by High Poisson and Low Periodic in Fig. 14, respectively. For each protocol and traffic model, we run our testbed more than 10 hours with 5 repetitions, where all experiments have been performed during 10 pm to 8 am to minimize external factors such as the effect of human and Wi-Fi activities. Fig. 13 shows two snapshots of routing topologies made by SR-0 and SEDA-Net, clearly exhibiting how SEDA-Net helps in generating well-balanced routing topology using self-configuration. Fig. 14 shows the *virtual* lifetime (i.e., until its predefined initial energy is depleted) and reliability (i.e., PRR) under both traffic conditions. We observe that under both High Poisson and Low Periodic, SEDA-Net outperforms all other protocols, where the lifetime increases by up to 59 percent (4.4 hours of LR-0 is increased to 7 hours of ours) and 113 percent (6.51 hours of WR-0 is improved to 13.84 hours of ours), respectively. Under Low Periodic, SEDA-Net shows slightly longer lifetime than LR-LTMAX. This is because densely deployed Wi-Fi APs in our office building incurs poor link qualities of SR. Thus, the advantage of utilizing SR is limited under low traffic, which is evidenced by the result that the performances of both SR and WR are significantly lower than the others. PRR of all protocols are shown in Fig. 14 as black lines. While SR and WR protocols show poor performance due to the interference of Wi-Fi especially for nodes in non Line-of-Sight, SEDA-Net can achieve highly reliable data delivery up to 98 percent by adaptively configuring to the given environment.

Dynamic Case. In contrast to the static case, we also run our testbed in the daytime with human and Wi-Fi activities.



(a) Node deployment in our office (b) Lifetime and end-to-end delay

Fig. 15. Daytime evaluation for dynamic case.

Fig. 15a presents our office room with 20 nodes' deployment. To perform an experiment in the office room, we configure the short-range and long-range radios' transmission power with -3 dBm and -6 dBm, respectively; thus, a multi-hop network is built by SEDA-Routing. In this evaluation, we consider a bursty traffic application where data packets are generated 10 packets per second for each node with a random interval ranging over 10 - 60 seconds. Fig. 15b demonstrates the virtual lifetime and average end-to-end delay (i.e., the delay from data generation to data arrival at the base station, averaged over all nodes). In the dynamic case, due to severe Wi-Fi interference, SEDA-Net configures the network mainly by the long-range radio, which is evidenced by the longer lifetime of LR-LT compared to others. More precisely, SEDA-Net achieves 0.89 hours of the virtual lifetime while SR-LT, LR-LT, and WR-LT have 0.57, 0.765, and 0.61 hours, respectively. In terms of the end-to-end delay, under bursty traffic, the low data rate of the long-range radio incurs high delay of 1.51 seconds for LR-LT. By leveraging the short- and long-range radios properly for the given environment, SEDA-Net minimizes the average delay of 1.34 seconds on average, where the protocols, based on the short-range radio, demonstrate shorter delay, but high variances due to external interference.

6 Conclusion

We proposed a self-configuring protocol with dual-radio sensors, called ***SEDA-Net***. SEDA-Net is designed to tackle the energy efficiency problem for Low-power and Lossy Networks (LLNs) with a cross-layer approach. SEDA-Net is composed of ***SEDA-MAC***, ***SEDA-Routing***, and ***Cross-Opt*** which adaptively determine the best configuration in a joint of MAC and network layers. By leveraging the characteristics of dual radios, SEDA-Net finds the optimized configuration which approximates the NP-Complete problem of the network lifetime maximization. We validated the performance of SEDA-Net using extensive simulations and real testbed experiments, demonstrating the lifetime improvement of up to 113 percent.

Acknowledgments

This work was supported by the Institute for Information and Communications Technology Planning and Evaluation (IITP) Grant funded by Korea government (MSIT) under Grant 2018-0-00772, for the development of an ultra-low power (500uW) long range radio for ultra-small IoT based on new Two-Tone OOK modulation scheme, and BK21 FOUR project of the National Research Foundation of Korea Grant.

REFERENCES

[1] IEEE Standard for Local and Metropolitan Area Networks-Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), *IEEE Standard 802.15.4–2011*, 2011. [Online]. Available: https://ieeexplore.ieee.org/servlet/opac?punumber=6012485

[2] IEEE Standard for Local and Metropolitan Area Networks-Part 15.4g: Low-Rate Wireless Personal Area Networks (LR-WPANs)," *IEEE Standard 802.15.4g/D7*, 2012. [Online]. Available: https://ieeexplore.ieee.org/servlet/opac?punumber=6178215

[3] Firefly Mote. Accessed: Dec. 2017. [Online]. Available: http://zolertia.io/product/hardware/firefly

[4] Waspmote. Accessed: Aug. 2017. [Online]. Available: http://www.libelium.com/products/waspmote

[5] OpenMote B. Accessed: Feb. 2018. [Online]. Available: http://www.openmote.com/product/openmote-b-single

[6] M. DelPrete, D. Masotti, A. Costanzo, M. Magno, and L. Benini, "A 2.4 GHz-868 MHz dual-band wake-up radio for wireless sensor network and IoT," in *Proc. 11th Int. Conf. Wirel. Mobile Comput., Netw. Commun.*, 2015, pp. 322–328.

[7] D. Spenza, M. Magno, S. Basagni, L. Benini, M. Paoli, and C. Petrioli, "Beyond duty cycling: Wake-up radio with selective awakenings for long-lived wireless sensing systems," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 522–530.

[8] J. Oller, I. Demirkol, J. Casademont, J. Paradells, G. U. Gamm, and L. Reindl, "Has time come to switch from duty-cycled MAC protocols to wake-up radio for wireless sensor networks?" *IEEE/ACM Trans. Netw.*, vol. 24, no. 2, pp. 674–687, Apr. 2016.

[9] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proc. 4th Int. Conf. Embedded Netw. Sensor Syst.*, 2006, pp. 307–320.

[10] 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 *Proc. 6th ACM Conf. Embedded Netw. Sensor Syst.*, 2008, pp. 1–14.

[11] D. Kim, J. Jung, Y. Koo, and Y. Yi, "Revisiting sensor MAC for periodic monitoring: Why should transmitters be early birds?" in *Proc. 14th Annu. IEEE Int. Conf. Sensing, Commun., Netw.*, 2017, pp. 1–9.

[12] R. C. Carrano, D. Passos, L. C. Magalhaes, and C. V. Albuquerque, "Survey and taxonomy of duty cycling mechanisms in wireless sensor networks," *IEEE Commun. Surveys Tut.*, vol. 16, no. 1, pp. 181–194, Jan.-Mar. 2014.

[13] N. S. Mazloum and O. Edfors, "DCW-MAC: An energy efficient medium access scheme using duty-cycled low-power wake-up receivers," in *Proc. Veh. Technol. Conf., VTC Fall*, 2011.

[14] N. S. Mazloum and O. Edfors, "Influence of duty-cycled wake-up receiver characteristics on energy consumption in single-hop networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3870–3884, Jun. 2017.

[15] J. Ansari, X. Zhang, and P. Mahonen, "Multi-radio medium access control protocol for wireless sensor networks," *Int. J. Sensor Netw.*, vol. 8, no. 1, pp. 47–61, 2010.

[16] M. Doddavenkatappa, M. C. Chan, and A. L. Ananda, "A dual-radio framework for MAC protocol implementation in wireless sensor networks," in *Proc. IEEE Int. Conf. Commun.*, 2011, pp. 1–6.

[17] F. Z. Djiroun and D. Djenouri, "MAC protocols with wake-up radio for wireless sensor networks: A review," *IEEE Commun. Surveys Tut.*, vol. 19, no. 1, pp. 587–618, Jan.-Mar. 2017.

[18] RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks, IETF RFC6550, 2012. [Online]. Available: https://tools.ietf.org/html/rfc6550

[19] A. Dunkels, B. Grönvall, and T. Voigt, "Contiki-a lightweight and flexible operating system for tiny networked sensors," in *Proc. 29th Annu. IEEE Int. Conf. Local Comput. Netw.*, 2004, pp. 455–462.

[20] P. Thubert, "Objective function zero for the routing protocol for low-power and lossy networks (RPL)," *IETF RFC6552*, 2012.

[21] C. Gu, R. Tan, X. Lou, and D. Niyato, "One-hop out-of-band control planes for low-power multi-hop wireless networks," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, 2018, pp. 1187–1195.

[22] "LoRa alliance. Accessed: Mar. 2015. [Online]. Available: https://www.lora-alliance.org/

[23] G. S. Luz, L. F. Vieira, M. A. Vieira, and O. Gnawali, "DCTP-A and DCTP-I: Collection tree protocols for dual radio platforms," in *Proc. 22nd International ACM Conf. Model, Anal. Simul. Wirel. Mobile Syst.*, 2019, pp. 87–94.

[24] Y. Wu, S. Fahmy, and N. B. Shroff, "On the construction of a maximum-lifetime data gathering tree in sensor networks: NP-completeness and approximation algorithm," in *Proc. IEEE INFOCOM 27th Conf. Comput. Commun.*, 2008, pp. 356–360.

[25] J. Liang, J. Wang, J. Cao, J. Chen, and M. Lu, "An efficient algorithm for constructing maximum lifetime tree for data gathering without aggregation in wireless sensor networks," in *Proc. IEEE INFOCOM*, 2010, pp. 1–5.

[26] T.-W. Kuo and M.-J. Tsai, "On the construction of data aggregation tree with minimum energy cost in wireless sensor networks: NP-completeness and approximation algorithms," in *Proc. IEEE INFOCOM*, 2012, pp. 2591–2595.

[27] F. Boubekeur, L. Blin, R. Leone, and P. Medagliani, "Bounding degrees on RPL," in *Proc. ACM Symp. QoS Secur. Wirel. Mobile Netw.*, 2015, pp. 123–130.

[28] O. Gaddour, A. Koubâa, N. Baccour, and M. Abid, "OF-FL: QoS-aware fuzzy logic objective function for the RPL routing protocol," in *Proc. IEEE 12th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wirel. Netw.*, 2014, pp. 365–372.

[29] H.-S. Kim, H. Kim, J. Paek, and S. Bahk, "Load balancing under heavy traffic in RPL routing protocol for low power and lossy networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 4, pp. 964–979, Apr. 2017.

[30] O. Iova, F. Theoleyre, and T. Noel, "Improving the network lifetime with energy-balancing routing: Application to RPL," in *Proc. IEEE 7th IFIP Wirel. Mobile Netw. Conf.*, 2014, pp. 1–8.

[31] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. 10th Annu. Int. Conf. Mobile Comput. Netw.*, 2004, pp. 114–128.

[32] M. Alicherry, R. Bhatia, and L. E. Li, "Joint channel assignment and routing for throughput optimization in multi-radio wireless mesh networks," in *Proc. 11th Annu. Int. Conf. Mobile Comput. Netw.*, 2005, pp. 58–72.

[33] M. Genetzakis and V. A. Siris, "A contention-aware routing metric for multi-rate multi-radio mesh networks," in *Proc. IEEE 5th Annu. IEEE Commun. Soc. Conf. Sensor, Mesh Ad Hoc Commun. Netw.*, 2008, pp. 242–250.

[34] S. Nirjon, A. Nicoara, C.-H. Hsu, J. Singh, and J. Stankovic, "Multinets: Policy oriented real-time switching of wireless interfaces on mobile devices," in *Proc. IEEE 18th Real Time Embedded Technol. Appl. Symp.*, 2012, pp. 251–260.

[35] Q. Peng, M. Chen, A. Walid, and S. Low, "Energy efficient multipath TCP for mobile devices," in *Proc. 15th ACM Int. Symp. Mobile ad hoc Netw. Comput.*, 2014, pp. 257–266.

[36] A. U. Chaudhry, R. H. Hafez, and J. W. Chinneck, "On the impact of interference models on channel assignment in multi-radio multi-channel wireless mesh networks," *Ad Hoc Netw.*, vol. 27, pp. 68–80, 2015.

[37] S. Avallone and A. Banchs, "A channel assignment and routing algorithm for energy harvesting multiradio wireless mesh networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1463–1476, May 2016.

[38] X. Chen, Y. Xu, and A. Liu, "Cross layer design for optimizing transmission reliability, energy efficiency, and lifetime in body sensor networks," *Sensors*, vol. 17, no. 4, 2017, Art. no. 900.

[39] A. B. Ammar, A. Dziri, M. Terre, and H. Youssef, "Cross-layer approach based energy minimization for wireless sensor networks," *Wireless Pers. Commun.*, vol. 98, no. 2, pp. 2211–2221, 2018.

[40] J. Tan, A. Liu, M. Zhao, H. Shen, and M. Ma, "Cross-layer design for reducing delay and maximizing lifetime in industrial wireless sensor networks," *EURASIP J. Wireless Commun. Netw.*, vol. 2018, no. 1, 2018, Art. no. 50.

[41] S. Randhawa and S. Jain, "Cross-layer energy based clustering technique for heterogeneous wireless sensor networks," *Wirel. Pers. Commun.*, vol. 114, pp. 1207–1233, 2020.

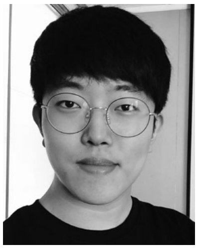
[42] Y. Chen and Q. Zhao, "On the lifetime of wireless sensor networks," *IEEE Commun. Lett.*, vol. 9, no. 11, pp. 976–978, Nov. 2005.

[43] I. Dietrich and F. Dressler, "On the lifetime of wireless sensor networks," *ACM Trans. Sensor Netw.*, vol. 5, no. 1, 2009, Art. no. 5.

[44] C. H. Papadimitriou, "The complexity of the capacitated tree problem," *Networks*, vol. 8, no. 3, pp. 217–230, 1978.

[45] J. Jung, J. Hong, and Y. Yi, "SEDA-Net source code." [Online]. Available:https://github.com/mp3jjk/SEDA-Net

Jinhwan Jung received the B.S. degree from the School of Electrical Engineering, Korea University, South Korea, in 2014 and the Ph.D degree from the School of Electrical Engineering, KAIST, South Korea, in 2020. His research interests include low-power networks and systems, IoT, and applied machine learning in networking.



Joonki Hong received the B.S. and M.S. degrees in electrical engineering, in 2015 and 2017, respectively, from KAIST, South Korea, where he is currently working toward the Ph.D. degree with the School of Electrical Engineering. His research interests include low-power network, IoT, and machine learning.

Yung Yi received the B.S. and M.S. degrees from the School of Computer Science and Engineering, Seoul National University, South Korea, in 1997 and 1999, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, The University of Texas at Austin, in 2006. From 2006 to 2008, he was a postdoctoral research associate with the Department of Electrical Engineering, Princeton University. He is currently a full professor at the Department of Electrical Engineering, KAIST, South Korea. His current research interests include the design and analysis of computer networking and wireless communication systems, especially congestion control, scheduling, and interference management, with applications in wireless ad hoc networks, broadband access networks, economic aspects of communication networks, and green networking systems. He was the recipient of the Best Paper Awards at the IEEE SECON 2013 and the ACM MobiHoc 2013, and the IEEE William R. Bennett Award, in 2016.