Reliable and Energy-Efficient Downward Packet Delivery in Asymmetric Transmission Power-Based Networks

HYUNG-SIN KIM and MYUNG-SUP LEE, Seoul National University
YOUNG-JUNE CHOI and JEONGGIL KO, Ajou University
SAEWOONG BAHK, Seoul National University

In low-power wireless networks, maintaining multihop connectivity is considered effective in constructing communication routes between individual nodes to a gateway. Since sensor networks are typically used for data collection, multihop routing protocols are designed to find routes optimal in upward directions. As sensor networks become widely applied to diverse applications, efficient downward traffic delivery also becomes important. To achieve this, we consider an asymmetric transmission power-based network (APN), where a power-supplied gateway uses high-power radios to cover the entire network via single-hop transmission, whereas common nodes use low-power transmissions. For effective APN operations, we propose a single-hop downlink protocol (SHDP) that consists of direct downlink transmission, local acknowledgment, neighbor forwarding, and contention resolution among the destination's neighbors. We evaluate SHDP through mathematical analysis, simulations, and testbed experiments. Our proposal outperforms other competitive multihop routing protocols. Specifically, SHDP shows high packet delivery performance and lowers the duty cycle greatly while reducing the packet transmission overhead by >50%.

CCS Concepts: • **Networks → Network design principles**

Additional Key Words and Phrases: Low power and lossy network, wireless sensor network, asymmetric power network, RPL, downward packet delivery, energy efficiency

ACM Reference Format:
Hyung-Sin Kim, Myung-Sup Lee, Young-June Choi, Jeonggil Ko, and Saewoong Bahk. 2016. Reliable and energy-efficient downward packet delivery in asymmetric transmission power-based networks. ACM Trans. Sen. Netw. 12, 4, Article 34 (September 2016), 25 pages.
DOI: http://dx.doi.org/10.1145/2983532

For H.-S. Kim, M.-S. Lee, and S. Bahk, this work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2015R1A2A2A01008240) and in part by the ICT R&D program of MSIP/IITP [B0717-16-0026].
For Y.-J. Choi, this research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (2015R1D1A1A01058025).
For J. Ko, this research was supported by a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (grant number: HI16C0982).
S. Bahk is the corresponding author.
An earlier version of this article appeared in the Proceedings of the IEEE International Conference on Communications (ICC'14), June 2014 [Kim et al. 2014].
Authors' addresses: H.-S. Kim, M.-S. Lee, and S. Bahk, Department of Electrical and Computer Engineering, Seoul National University, 1 Gwanak-ro, Gwanak-gu, Seoul, Korea 151-742; emails: {hskim, mslee}@netlab.snu.ac.kr, sbahk@snu.ac.kr; Y.-J. Choi and J. Ko, Department of Software and Computer Engineering, 206 Worldcup Ro, Yeongtong-Gu, Suwon, Korea 16499; emails: {choiyj, jgko}@ajou.ac.kr.
Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2016 ACM 1550-4859/2016/09-ART34 $15.00
DOI: http://dx.doi.org/10.1145/2983532

1. INTRODUCTION

Over the past decade, various wireless sensing systems have been designed, and one of the common choices in system design was to use a homogeneous radio module for all the nodes in the network. By doing so, wireless links could be considered to have "close to" symmetric performance. This symmetry enables researchers to easily apply techniques developed for various mobile ad hoc networks (MANETs) in the research domain of wireless sensor networks (WSNs). Furthermore, this symmetric property helped develop various low-power multihop routing protocols that operate under limited wireless capacity and energy for data delivery from sensor nodes to gateways that are typically distantly located.

Examples of such low-power multihop routing protocols include the collection tree protocol (CTP) [Gnawali et al. 2008] and the routing protocol for low-power and lossy networks (RPL) [Ko et al. 2011; Winter et al. 2012], which are widely applied in sensor network deployments [Ancillotti et al. 2013; Ceriotti et al. 2011; Ko et al. 2010]. These protocols mostly focus on serving uplink traffic where data reporting takes up a major portion. While downlink traffic is considered in protocols such as RPL, the efficiency is usually dependent on the uplink quality, since the reverse of them is used as the downlink.

However, as wireless sensing system applications become diverse, their traffic types also start to vary. Considering various application services, simply exploiting symmetric wireless links may not be considered a good design choice. For example, with an application that mostly generates downward traffic (e.g., from the gateway to sensor nodes), managing multihop routes at each node in dynamic wireless environments will result in a significant amount of control overhead. Moreover, multihop routing protocols naturally require each node to have a designated memory space to store and manage these routes [Clausen et al. 2011].

In this article, we address such inefficiencies in multihop routing protocols for applications with a high emphasis on downward traffic and propose an asymmetric transmission power network (APN). In the APN, a gateway node with a high power radio maintains *single-hop* connectivity to many low-powered nodes, which are deployed distantly for application-specific purposes. The gateway node is typically connected to a power source and takes the role of distributing information to the low-power nodes.

Nevertheless, in order to ensure reliable packet delivery in APNs, a low-powered destination node needs to send an acknowledgment (ACK) packet toward the gateway for each received packet in a multihop manner. This action of multihop ACK transmissions incurs severe uplink traffic and high energy consumption. As a way of achieving both reliable and energy-efficient packet delivery with the single-hop downlink connectivity, we propose a protocol, termed as the single-hop downlink protocol (*SHDP*) [Kim et al. 2014], which exploits the single-hop downward transmission capability of the gateway, local ACK exchange between the destination and its neighbors, and neighbor forwarding to ensure best-effort packet delivery.

In *SHDP*, the destination node is not required to send ACK packets to the gateway directly. Instead, *SHDP* achieves reliability by allowing the destination and its neighbors to exchange ACK packets locally, and then the neighbors forward any missing packets opportunistically upon detecting a transmission failure. As a result, each low-power node transmits an ACK packet toward its neighbors upon a successful packet reception destined for itself, and keeps track of neighbor-initiated ACK packets by overhearing the wireless channel. When a transmission failure for a node is detected (e.g., lack of ACK packet), neighbors participate in local retransmission after taking random backoffs.

In our testbed experiments, *SHDP* successfully achieves a 99.4% downward packet reception ratio (PRR) in dynamic wireless environments. While this PRR is only a 4% increase over the RPL routing protocol, *SHDP* reduces the relative transmission overhead by more than 50%, leading to a lower radio duty cycle. Furthermore, our experimental results show that the performance of the "worst-case" node is significantly improved with *SHDP*, implying that the fairness among nodes in the network can be achieved to provide high-quality networking services for the longest duration possible.

The contributions of this work are threefold:

—We consider applications that mainly generate downward traffic and quantify limitations of existing multihop routing protocols in delivering downward traffic.
—We design an APN system architecture where a gateway node uses a high-power radio in communicating with many low-power sensor nodes that are deployed over a wide geographical region. To ensure highly reliable and energy-efficient packet delivery, we propose *SHDP*, where the gateway delivers packets to sensor nodes via single-hop transmissions.
—Through mathematical analysis, simulations, and indoor testbed experiments, we evaluate *SHDP* against existing multihop routing protocols extensively, considering various channel and traffic conditions. By exploiting high transmission power of the gateway, *SHDP* can lower the transmission overhead of other low- power nodes by 50%.

While this work is an extension of our previous work [Kim et al. 2014], the technical contribution differs in several ways. First, we provide a feasibility analysis and introduce a number of application scenarios in which APNs can benefit the networking performance. Second, we redesign *SHDP* so that it interoperates with the RPL routing protocol, allowing our APN to support single-hop downward and multihop upward data transmissions. Third, we implement *SHDP* on real resource-constrained embedded devices and evaluate its performance using an indoor testbed under various traffic scenarios. Lastly, we present a mathematical analysis on the packet delivery ratio and latency performance.

The rest of this article is structured as follows. In Section 2, we introduce the concept of APN and present applications that can benefit from asymmetric transmission power capabilities of nodes. We propose an APN architecture in Section 3 and provide its analytical evaluations in Section 4. Section 5 shows our simulation results, and Section 6 presents the results obtained from the indoor testbed implementation. We conclude with a discussion of our results in Section 8.

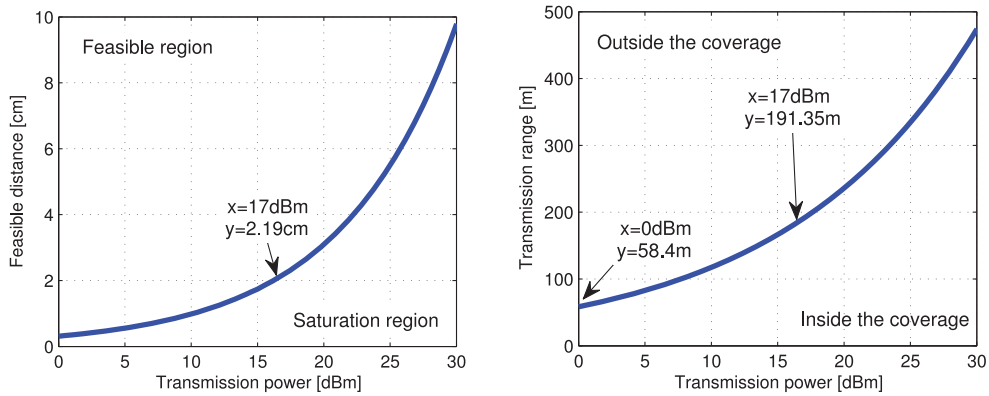
2. ASYMMETRIC TRANSMISSION POWER NETWORKS AND THEIR APPLICATIONS

Before discussing the details of *SHDP*, we start with introducing the APN architecture and consider some applications that can benefit from utilizing the APN architecture. This work targets a specific APN architecture where the gateway node uses high-power transmissions and energy-limited sensor nodes operate with low-power constraints. While the hardware configuration of a sensor node is simple, the heterogeneity between the gateway and common sensor nodes requires changes in the protocol design.

2.1. APN Characteristics

Assume that $PL(d)$ represents the path loss of a channel as a function of the distance d between a sender and a receiver. Let us denote the transmission power at the gateway and the received power at a receiver as P and p_r, respectively. Then, we have

$$p_r = \frac{P}{PL(d)}. \tag{1}$$



(a) Feasible distance according to the transmission power (b) Transmission range according to the transmission power

Fig. 1. Enabling factor for an APN.

Representing the bit error rate (BER) according to the received signal-to-noise ratio (SNR) γ as $BER(\gamma)$, we can express the packet error rate PER as

$$PER(\gamma) = 1 - (1 - BER(\gamma))^B, \tag{2}$$

where B is the packet size in bits, $\gamma = p_r/n_0$, and n_0 is the magnitude of noise. Denoting a given PER requirement and minimum received power required to meet the receiver sensitivity as e_{th} and $p_{r,th}$, respectively, we have $PER(\gamma_{th}) = e_{th}$, where $\gamma_{th} = p_{r,th}/n_0$.

We use high transmission power for a WSN, which mainly considers low-power communications. This requires checking whether low-power transceivers, widely used in WSNs (e.g., CC2420 [Texas Instruments 2006]), can accept high-power packet transmissions without receiver malfunctioning caused by power saturation. Let us denote the receiver saturation power as p_{sat}, which indicates that the receiver malfunctions when its receiving power is greater than p_{sat} (i.e., receiver saturation). Then, the condition for the receiver to decode a packet from the gateway successfully satisfies

$$p_{r,th} \leq p_r \leq p_{sat}. \tag{3}$$

We define the feasible transmission distance R_{fs} and the transmission range R_{tx} as

$$R_{fs} = \min\{d \,|\, p_r \leq p_{sat}\}, \tag{4}$$

$$R_{tx} = \max\{d \,\big|\, p_r \geq p_{r,th}\}. \tag{5}$$

Figure 1 illustrates R_{fs} and R_{tx} according to the transmission power P. In this example, we assume that p_{sat} = 10dBm following the data sheet of CC2420 [Texas Instruments 2006]. The values of $e_{th} = 0.1$ and $p_{r,th}$ = -87dBm have been confirmed through extensive experiments using CC2420 [Chen and Terzis 2011; Srinivasan et al. 2010]. In addition, we assume that $PL(d)$ follows the indoor path loss model of the IEEE 802.15.4 standard [IEEE Std 802.15.4-2003 2003] given by

$$PL(d) = \begin{cases} 10^{4.02+2\log(d)}, & \text{for } d \leq 8 \\ 10^{5.85+3.3\log(d/8)}, & \text{otherwise.} \end{cases} \tag{6}$$

We consider transmission power in the range from 0dBm to 30dBm in accordance with the maximum transmission power of CC2420 [Texas Instruments 2006] and the FCC regulations for the 2.4GHz band [Federal Communications Commision 1993].

Figure 1(a) shows that the feasible distance is negligibly small since it is shorter than 10cm even when the maximum transmission power is applied. Figure 1(b) shows

that the transmission range of the gateway that uses the Wi-Fi transmission power of 17dBm [IEEE Std 802.11-2009 2009] (i.e., ∼191 meters) is 3.3 times longer than that of a low-power node (i.e., ∼58 meters).

These results confirm that high-power transmissions at the gateway can connect a wide area via a single hop, while not power saturating the bandwidth for low-power transceivers. Overall, this result validates the applicability of APNs in low-power WSN deployments. We emphasize that, although this physical feasibility check does not *guarantee* the usefulness of APNs in practice, it is necessary to have a solid motivation before designing an APN-based protocol.

2.2. APN Applications

Compared to conventional WSNs in which all nodes use the same transmission power, the main benefit of an APN is in providing single-hop delivery for downward traffic without configuring a multihop downward route. To follow are several applications that can take advantage of the APN architecture.

—**Electronic Price Management:** Product prices at supermarkets and retail stores may change frequently depending on the time of sale and prices at competing vendors. Currently, most of the retagging processes are done manually, which results in added labor cost and frequent customer claims due to mis-tagging [Aruba Networks 2012]. Electronic price tagging systems send a unicast packet containing updated price information to each target node (i.e., price tag with a radio module). Within an APN, price updates can be performed with minimal effort by broadcasting through a high-power gateway node.

—**Emergency Message Broadcasting:** Various applications are designed to alert inhabitants in a target environment where emergency situations arise [Lorincz et al. 2004]. To achieve rapid delivery of emergency messages, flooding can be employed, which requires that all nodes participate in forwarding [Marot et al. 2004; Werner-Allen et al. 2006]. The resultant multihop latency, coupled with duty cycling of nodes to conserve energy, can degrade overall system performance [Wang and Liu 2009; Yildirim and Kantarci 2014]. APN provides an efficient alternative for propagating alert messages.

—**Mobile Applications:** With emerging cyber physical systems (CPSs) and the Internet of Things (IoT), mobility becomes an integral part of the low-power wireless networks. These applications include robotic networks and military-related sensing applications. While various network protocols have been proposed and designed, frequent loss of link connectivity due to node mobility makes the bidirectional route establishment challenging [Fink et al. 2012]. In APNs, mobile nodes need to keep their routing entries only toward the gateway, since downward transmissions are completed using a high-power single-hop link.

While the concept of APN is not new, design of an effective network architecture and performance evaluations in this work introduce new challenges when compared to traditional single- or multihop networks. Next we introduce our *SHDP* scheme, which aims to provide an efficient best-effort packet delivery for APN architectures.

3. DESIGN OF SHDP

3.1. SHDP Overview

In the proposed *SHDP*, each node (except the gateway) is responsible for the local ACK, neighbor forwarding, and forwarding contention mitigation processes. In a multihop environment with a high-power transmitting gateway, an ACK from a low-power destination node is typically unable to reach the gateway directly over single-hop

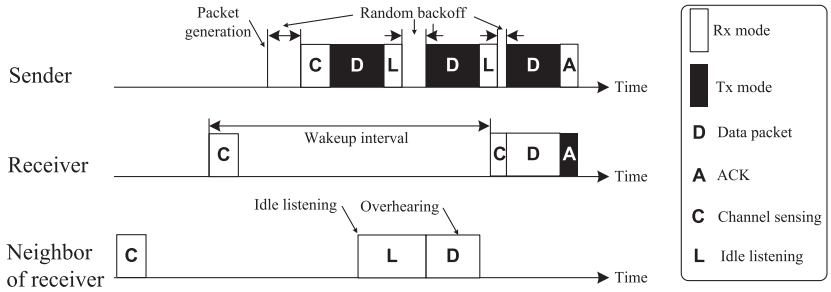


Fig. 2. An example of BoX-MAC-2 operation. BoX-MAC-2 achieves asynchronous packet delivery by exploiting periodic wakeup at the receiver and repetitive packet transmission at the sender, incurring a tradeoff between a sender and a receiver in energy consumption according to the wakeup interval.

transmission due to the physical distance. The problem is that without ACK reception, the gateway cannot confirm whether the packet was delivered successfully.

It can be the case that the destination node transmits ACKs to the gateway over multiple hops. However, this can take a long time to reach the gateway and incur significant traffic overhead when issued for all data packets. To address this issue while maintaining a low networking overhead, we propose to use local-ACKs, which confirm packet delivery using the help of neighbors near the destination. For local-ACKs, the destination's neighbors overhear the downlink packet transmissions from the gateway and also the (single-hop-transmitted) ACK from the destination node if the packet is successfully delivered. If a transmission failure occurs and an ACK cannot be overheard from the destination node, these neighbors who overhear the data packet from the gateway retransmit the packet for the destination node. To minimize the contention from this retransmission process as well, if a neighbor node realizes that a different neighbor already retransmitted the data packet (again from overhearing), the retransmission is suppressed.

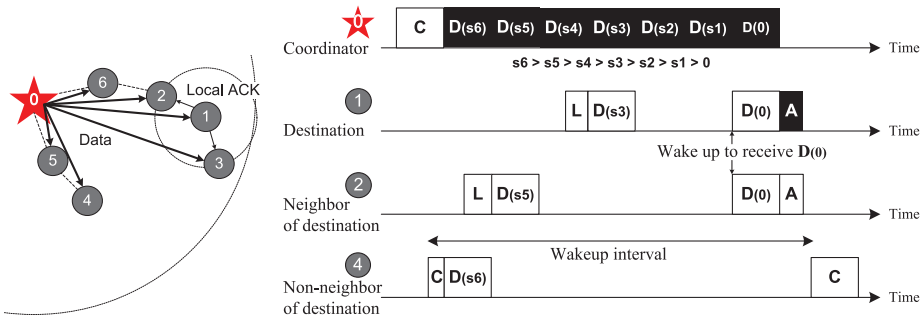
Through such a procedure, the gateway can deliver packets to destinations reliably despite failures in direct downlink transmissions. As a result of local-ACKs, *SHDP* prevents the network from making multihop downward transmissions (with a high-power root) and also prevents ACK packets from traveling over multiple upward hops to reach the gateway.

3.2. SHDP Architecture and Implementation

3.2.1. Baseline MAC Protocol. The design of *SHDP* is based on BoX-MAC-2 [Moss and Levis 2008], which is the default MAC protocol of TinyOS and is widely used in various WSN systems [Landsiedel et al. 2012; Puccinelli et al. 2012]. Figure 2 illustrates an example of the asynchronous BoX-MAC-2 operations. Here, each node periodically wakes up and checks for the channel-sensing period on whether the channel is busy or not. At this point, if the node identifies no traffic on the channel, it goes back to sleep. In case there are activities on the wireless channel, the node confirms if the packet is intended to itself, and, if so, it receives that packet, sends an ACK, and goes back to sleep to continue its periodic wakeup process.

A node with a packet to send performs channel sensing after a random backoff t_{bo} ($0 \leq t_{bo} \leq t_{bo,max}$). If the channel is busy, it accesses the channel again after waiting for an extended backoff time. If the channel is free, it repeats the process of random backoff and packet transmissions continuously until an ACK is received. In other words, the data packet is repetitively transmitted during a full wakeup interval.

Therefore, the channel-sensing period of the receiver should be longer than or equal to the maximum interval between repetitive packet transmissions at the sender (i.e.,



(a) Network topology (b) Time operation

Fig. 3. An example of direct transmission in *SHDP*. A forwarder node can opportunistically retransmit an overheard downward packet to increase network reliability. To mitigate contention, each potential forwarding node suppresses forwarding when detecting an ongoing forwarding process of the same packet by another node.

($t_{ack} + t_{bo,max}$), where t_{ack} is the ACK length) in order to allow the receiver to detect whether the channel has an ongoing transmission or not.

In Figure 2, the receiver node receives and acknowledges the packet in the second wakeup period. As we can see in the "neighbor of receiver" case in Figure 2, when a node starts listening to the wireless channel in the middle of a transmission, it continues to turn on its radio until the next repetitive transmission begins. As a result, the maximum idle listening period becomes ($t_{packet} + t_{ack} + t_{bo,max}$), where t_{packet} is the length of the data packet.

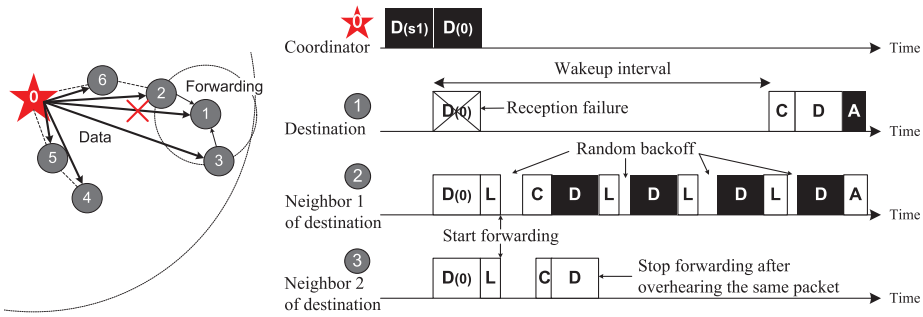
This repetitive transmission of a sender and the periodic wakeup of a receiver leads to a successful packet delivery without synchronization. If the sender fails to receive an ACK, it tries again to transmit the packet up to a maximum number of n_{tx} times.[1]

BoX-MAC-2 has a tradeoff relation between a sender and a receiver in energy consumption based on the wakeup interval. As the wakeup interval increases, the sender consumes more energy due to the increase in the number of repetitive packet transmissions, while the receiver consumes less energy owing to the reduced frequency in channel sensing. The proposed *SHDP* is designed to have low transmission overhead by eliminating multihop transmission for downward traffic.

3.2.2. Gateway's Direct Transmissions. We aim to achieve reliable downward packet transmissions by allowing a destination's neighbor nodes (rather than the gateway) to confirm the packet delivery by receiving ACK packets from the destination node. However, BoX-MAC-2 cannot support ACK exchange between the destination and its neighbor nodes since each node has no knowledge of when its neighbor nodes will be awake (or asleep). To alleviate the problem, we design a new transmission mechanism for the gateway, allowing nodes to wake up simultaneously and exchange local ACKs.

We describe the operation of our high-transmission power gateway using Figures 3 and 4, which show examples of a gateway's successful direct transmission and neighbor forwarding, respectively. The gateway node "0" transmits a packet to the destination with a high transmission power. To allow the randomly waking-up destination node to receive the packet, the gateway repetitively transmits the same packet for the entire wakeup interval. Of course, due to the asymmetric nature of APNs, the gateway does not expect to receive an ACK from the destination.

[1]One packet retransmission comprises multiple repetitive transmissions during a single wakeup interval.



(a) Network topology (b) Time operation

Fig. 4. An example of neighbor forwarding in *SHDP*. A forwarder node can opportunistically retransmit an overheard downward packet to increase network reliability. To mitigate contention, the (potential) forwarder node suppresses forwarding when detecting an ongoing forwarding process of the same packet by another node.

To support local ACK exchange, the gateway includes a time index s for each repetitive transmission for the same packet. The index s represents the time interval between the start of the current repetitive transmission to the end of the full repetitive transmission batch. To set s, we use a timer designed to manage repetitive transmissions in BoX-MAC-2. This timer runs for the wakeup interval t_{wakeup}. The repetitive transmission process starts with the timer and ends when it expires. For each repetitive transmission, the gateway sets the index s as the currently remaining time for the timer to stop and inserts s into the packet header. The last repetitive packet transmission has $s = 0$. To minimize the idle listening period of a sensor node, the gateway does not use a random backoff between two repetitive packet transmissions (i.e., $t_{bo} = 0$).

3.2.3. Local ACK Exchange. Due to the use of time index s in the packet header from the gateway, the destination node can decide to exchange local ACKs with its neighbor nodes as follows. When the destination wakes up and receives a packet, it first checks the packet's destination address. If the node is the packet's destination (node 1 in Figures 3 and 4), the node checks whether $s = 0$ (i.e., the end of the gateway's repetitive packet transmission process). If $s > 0$, the destination sleeps to save energy and wakes up when $s = 0$ is expected. The sleep interval here $t_{sleep}(s)$ is obtained by the interval between the end time of the current packet transmission duration and the start time of the last packet to be transmitted (i.e., $s = 0$)[2]. This sleep interval can be represented as

$$t_{sleep}(s) = s - t_{packet}. \tag{7}$$

Upon receiving the $s = 0$ packet, the destination broadcasts an ACK for its neighbors to overhear. Thus, the main difference between our proposal and the baseline BoX-MAC-2 is that the destination waits until it receives the latest of the sender's repetitive transmissions before issuing an ACK packet.[3]

If a node is not the packet's intended destination, the node checks whether the destination of the packet is one of its neighbors. If so, the node operates identically to

[2]A small guard time is added in real implementations due to practical issues such as clock drift and per-node processing delays [Pak et al. 2008].

[3]With the sequence number, we can allow the destination and its neighbors to simultaneously wake up for receiving packet $s = 0$ and stay awake until the exchange of an ACK. We can also configure the destination to send an ACK when receiving the first packet with $s > 0$, which allows neighbor nodes who receive this ACK (early) to go into sleep mode as a way of conserving power.

the destination node up to the point when it receives packet $s = 0$. Once receiving the last packet, the node listens to the channel to overhear the ACK from the destination and goes to sleep once receiving the ACK (nodes 2 and 3 in Figure 3).

In the case in which the node is neither the destination nor one of the destination's neighbors, the node puts to sleep its radio for the entire wakeup interval (e.g., nodes 4–6 in Figure 3). Therefore, these nodes conserve their energy in the meantime.

3.2.4. Neighbor Forwarding. As we show in Figure 4, when the destination fails to receive the last packet transmission from the gateway, neighbor nodes detect transmission failure via a missing local ACK and try to forward the received packet. The neighbor forwarding procedure is the same as a normal transmission of our baseline MAC protocol (BoX-MAC-2). That is, a neighbor node forwards the packet repetitively using a random backoff mechanism, but without using the time index s. From this, the destination node realizes that the packet is a "forwarded packet" from one of its neighbors. Once received at the final destination, an ACK exchange takes place. When failing to receive an ACK from the destination (for the neighbor-forwarded packet), the forwarder node retransmits the packet for a maximum number of n_{tx} times, resulting in a best-effort approach.

If multiple neighbors attempt packet forwarding simultaneously, a contention issue occurs [Choi et al. 2006]. To resolve this, a forwarding node carrier senses for ongoing transmissions before attempting the packet forwarding. Once a neighbor realizes that a different node has already started a packet-forwarding attempt (e.g., via overhearing before its transmissions), the node suppresses its message transmissions under the assumption that the other node (already occupying the channel) will perform its best effort in delivering the message. This simple forwarding suppression scheme mitigates a large portion of contention that can occur in the neighbor-forwarding phase.

4. PERFORMANCE ANALYSIS

In this section, we analyze *SHDP* with respect to the packet reception ratio, latency, and power consumption. Consider a network of N_{tot} nodes that are randomly distributed over a given area and ready to receive downward packets from the gateway that is located at the center. Assume that each node has transmission range r and average number of neighbor nodes N_{ne} within the transmission range. We also assume that each node can receive packets transmitted from another node within r and suffers from PER $e\,(\leq e_{th})$. For mathematical tractability, we assume that each node, including the gateway, transmits a packet without contention, collision, and queuing delay. Thus, the analysis is for light traffic conditions with the general case addressed in the experimental performance evaluation.

In *SHDP*, a node does not know when the gateway starts its transmission, so we assume the time index s of the packet transmitted from the gateway is uniformly randomly distributed between 0 and t_{wakeup}, in the perspective of each node. The probability that the first received packet is the last one (i.e., $s = 0$) becomes small as the wakeup interval increases. This means that, in a low duty-cycle network, the destination and its neighbor nodes are likely to receive two repetitive packet transmissions from the gateway before local ACK exchange.

As a point of reference, we consider an ideal multihop downlink protocol (MHDP). In MHDP, the gateway is assumed to know a downward (multihop) route with minimum hop to each sensor node without running route discovery. Since MHDP is assumed to have no control overhead, it achieves the best performance compared to conventional multihop routing protocols. We compare *SHDP* against this optimal MHDP.

4.1. Packet Reception Ratio

Let $d_{hop}(k)$ be the hop distance between the gateway and node k when the gateway uses low power, which is given by $d_{hop}(k) = d(k)/r$, where $d(k)$ is the distance between the gateway and node k.

Assuming that n_{tx} is the maximum allowed number of retransmissions, the packet reception ratio (PRR) over one-hop transmission, PRR_{one}, is given by $PRR_{one} = 1 - e^{n_{tx}}$. Then the PRR at node k in MHDP becomes

$$PRR_{MHDP}(k) = PRR_{one}^{d_{hop}(k)}, \tag{8}$$

which shows that $PRR_{MHDP}(k)$ exponentially decreases with $d_{hop}(k)$.

Unlike MHDP, all the nodes in *SHDP* are within one hop from the gateway. Thus, the PRR in *SHDP* is the same for all the nodes, which is given by

$$PRR_{SHDP} = PRR_{direct} + PRR_{indirect}. \tag{9}$$

PRR_{direct} is the probability that a packet from the gateway is delivered in one-hop transmission, that is, $PRR_{direct} = 1 - e$. $PRR_{indirect}$ represents the probability that the packet is indirectly delivered. Indirect packet delivery happens when the destination failed to receive the gateway's transmission but at least one of its neighbors successfully overheard two repetitive packet transmissions from the gateway. Hence, its probability is $1 - (1 - (1 - e)^2)^{N_{ne}}$. By mitigating forwarding contention, a forwarding neighbor accesses the medium and transmits the packet to the destination successfully with probability PRR_{one}. Thus, $PRR_{indirect}$ can be written as

$$PRR_{indirect} = e\left(1 - (1 - (1 - e)^2)^{N_{ne}}\right)PRR_{one}. \tag{10}$$

Obviously, indirect transmission significantly enhances reliability when N_{ne} is large. In a fully connected network, each node has at least one neighbor node, which implies

$$PRR_{indirect} \geq e\,(1 - e)^2\left(1 - e^{n_{tx}}\right). \tag{11}$$

This shows that *SHDP* can achieve reliable packet delivery with high probability through local ACK exchange and neighbor forwarding without the gateway's confirmation.

4.2. Latency

A successful packet transmission incurs average latency $t_{wakeup}/2$ to receive an ACK successfully, which increases by t_{wakeup} whenever a retransmission happens. If a packet is delivered after having j transmissions, the latency is given as $t_{wakeup}(j - 0.5)$. Since the probability that a packet is transmitted j times for successful delivery is $(1 - e)e^{j-1}$, the average latency for each hop is given as

$$L_{one} = \frac{t_{wakeup}(1 - e)\sum_{j=1}^{n_{tx}}\left(j - \frac{1}{2}\right)e^{j-1}}{PRR_{one}}. \tag{12}$$

For packet delivery from the gateway to node k in MHDP, the average latency becomes

$$L_{MHDP}(k) = d_{hop}(k)\,L_{one}. \tag{13}$$

Notice that $L_{MHDP}(k)$ linearly increases with $d_{hop}(k)$ and t_{wakeup}.

In *SHDP*, the latency is the same for all the nodes. That is,

$$L_{SHDP} = \frac{PRR_{direct}L_{direct} + PRR_{indirect}L_{indirect}}{PRR_{SHDP}}. \tag{14}$$

Here L_{direct} is the latency when the destination node succeeds in receiving the direct transmission, denoted as $L_{direct} = t_{wakeup}/2$, and $L_{indirect}$ is the latency from the destination receiving the packet via neighbor forwarding. Furthermore, $L_{indirect} = L_{one} + t_{wakeup}$, where L_{one} is the forwarding latency. We can notice here that L_{SHDP} linearly increases with t_{wakeup}. In addition, with a high PRR_{direct}, *SHDP* will result in low latency, given that a lower $PRR_{indirect}$ can be expected and $L_{direct} < L_{indirect}$ due to the smaller number of transmission attempts.

4.3. Power Consumption

We analyze the basic characteristics of the baseline protocol first, followed by the power consumption performance of MHDP and *SHDP*.

4.3.1. Basic Protocol Analysis. Assume that p_{tx}, p_{rx}, p_{cs}, and p_{idle} represent the required power for transmission, reception, channel sensing, and backoff time (or idle time), respectively. Each node in BoX-MAC-2 periodically wakes up and consumes energy ε_{cs} for channel sensing, which is

$$\varepsilon_{cs} = p_{cs}(t_{ack} + t_{bo,\max}). \tag{15}$$

The total power consumption for checking the wireless medium, $p_{cs,tot}$, is given as

$$p_{cs,tot} = \frac{N_{tot}\varepsilon_{cs}}{t_{wakeup}}. \tag{16}$$

$p_{cs,tot}$ represents the basic power consumption with no packet transmission or reception.

A node that has a packet to send performs channel sensing first, which consumes ε_{cs}. The number of repetitive transmissions for successful and erroneous transmissions, $n_{tx,s}$ and $n_{tx,f}$, is given by

$$n_{tx,s} = \left\lceil \frac{t_{wakeup}}{2\left(t_{packet} + t_{ack} + t_{bo,\max}/2\right)} \right\rceil, \tag{17}$$

$$n_{tx,f} = \left\lceil \frac{t_{wakeup}}{t_{packet} + t_{ack} + t_{bo,\max}/2} \right\rceil \tag{18}$$

where $\lceil x \rceil$ is the smallest integer not less than x. The energy consumptions for successful and erroneous packet transmissions, $\varepsilon_{tx,s}$ and $\varepsilon_{tx,f}$, are given by

$$\varepsilon_{tx,s} = \varepsilon_{cs} + n_{tx,s}\varepsilon_{tx}, \tag{19}$$

$$\varepsilon_{tx,f} = \varepsilon_{cs} + n_{tx,f}\varepsilon_{tx}. \tag{20}$$

ε_{tx} is the energy consumption for each (repetitive) transmission, which is given by

$$\varepsilon_{tx} = \frac{p_{idle}t_{bo,\max}}{2} + p_{tx}t_{packet} + p_{rx}t_{ack}, \tag{21}$$

since each (repetitive) transmission period comprises random backoff, packet transmission, and ACK reception. Ignoring the collision factor, we obtain the total energy consumed for a one-hop transmission (including retransmission) as

$$\varepsilon_{tx,one} = (1-e)\sum_{i=0}^{n_{tx}-1}\left(\varepsilon_{tx,s} + i\varepsilon_{tx,f}\right)e^{i} + n_{tx}\varepsilon_{tx,f}e^{n_{tx}}. \tag{22}$$

$\varepsilon_{tx,one}$ increases with t_{wakeup} like $\varepsilon_{tx,s}$ and $\varepsilon_{tx,f}$.

We now consider the power consumption of a receiver. When a receiver wakes up and performs channel sensing while the channel has an ongoing transmission, it can detect packet transmission but cannot decode it. Thus, its idle listening should last until the next repetitive transmission begins. This results in the average time for idle listening of

$$t_{listen} = \frac{(t_{packet} + t_{ack})(t_{bo,\max} + t_{packet}/2 + t_{ack}/2)}{t_{bo,\max} + t_{packet} + t_{ack}}. \tag{23}$$

The total energy consumed until the receiver finishes its one-hop reception is

$$\varepsilon_{rx,one} = p_{tx} t_{ack}(1 - e^{n_{tx}}) + p_{rx}(t_{packet} + t_{listen}) \left\{ (1-e) \sum_{i=1}^{n_{tx}} i e^{i-1} + n_{tx} e^{n_{tx}} \right\}. \tag{24}$$

If a sender transmits a packet, its neighbor nodes overhear it. Since a successful packet transmission lasts for the period of $\frac{t_{wakeup}}{2}$ on average, each neighbor of the sender can overhear the packet with probability 0.5. For an erroneous packet transmission, all the neighbors can overhear it. Thus, the energy consumptions required for overhearing successful and erroneous packet transmissions, denoted as $\varepsilon_{oh,s}$ and $\varepsilon_{oh,f}$, respectively, can be given as

$$\varepsilon_{oh,s} = \frac{p_{rx} N_{ne}(t_{packet} + t_{listen})}{2}, \tag{25}$$

$$\varepsilon_{oh,f} = p_{rx} N_{ne}(t_{packet} + t_{listen}). \tag{26}$$

The total energy consumption for overhearing a one-hop transmission becomes

$$\varepsilon_{oh,one} = (1-e) \sum_{i=0}^{n_{tx}-1} (\varepsilon_{oh,s} + i\varepsilon_{oh,f}) e^{i} + n_{tx} \varepsilon_{oh,f} e^{n_{tx}}. \tag{27}$$

4.3.2. MHDP Analysis. For MHDP, when a gateway transmits a packet to node k, $N_{hop}(k)$ senders and $N_{hop}(k)$ receivers are involved in packet delivery. We will ignore the energy consumption of the gateway since it is assumed to be connected to the power supply. Then the total energy consumption for $(N_{hop}(k) - 1)$ senders and $N_{hop}(k)$ receivers, denoted as $E^{tx}_{MHDP}(k)$ and $E^{rx}_{MHDP}(k)$, respectively, can be expressed as

$$E^{tx}_{MHDP}(k) = \varepsilon_{tx,one} \sum_{i=1}^{N_{hop}(k)-1} (1 - e^{n_{tx}})^{i}, \tag{28}$$

$$E^{rx}_{MHDP}(k) = \varepsilon_{rx,one} \sum_{i=0}^{N_{hop}(k)-1} (1 - e^{n_{tx}})^{i}. \tag{29}$$

Each node within the sender's transmission range overhears the packet transmission by incurring more energy consumption. The total energy consumption for overhearing a one-hop transmission while the gateway transmits a packet toward node k becomes

$$E^{oh}_{MHDP}(k) = \varepsilon_{oh,one} \sum_{i=0}^{N_{hop}(k)-1} (1 - e^{n_{tx}})^{i}. \tag{30}$$

Obviously, $E^{tx}_{MHDP}(k)$, $E^{rx}_{MHDP}(k)$, and $E^{oh}_{MHDP}(k)$ increase with $N_{hop}(k)$ since the number of relaying nodes increases. Nodes near the gateway consume more energy than those far from the gateway due to the heavy burden imposed by relaying.

Let us denote the packet interarrival time as t_{int}, and assume that the gateway generates packets toward each destination randomly. Then, the total power consumed by all the nodes becomes

$$P_{MHDP} = p_{cs,tot} + \frac{1}{t_{int}N_{tot}} \sum_{k=1}^{N_{tot}} \left(E^{tx}_{MHDP}(k) + E^{rx}_{MHDP}(k) + E^{oh}_{MHDP}(k)\right). \tag{31}$$

$\varepsilon_{tx,one}$ in $E^{tx}_{MHDP}(k)$ increases, $p_{cs,tot}$ decreases with t_{wakeup}, and the weight for $\varepsilon_{tx,one}$ increases with the network size. Thus, the MHDP is not appropriate for supporting low-duty-cycled large-scale WSNs efficiently.

4.3.3. SHDP Analysis. In *SHDP*, the energy consumption of all nongateway nodes is the same given that they are within single-hop range of the gateway and have equal chances for transmissions, receptions, and overhearings in the long term.

For each packet transmission from the gateway, while all nodes in the network overhear the packet, only neighbors of the destination node try to forward the overheard packets, and only one of them finally participates in the packet delivery as a forwarder node (i.e., the neighbor of the destination occupying the channel first for packet forwarding). The total energy consumption for a complete transmission follows:

$$E^{tx}_{SHDP} = (1-(1-e)^2)\left\{\begin{array}{l}\left(1-(1-(1-e)^2)^{N_{ne}}\right)\varepsilon_{tx,one}+\\ \left((1-e)^2N_{ne}-1\right)\left(\frac{p_{idle}t_{bo,\max}}{2}+\varepsilon_{cs}\right)\end{array}\right\}. \tag{32}$$

The first term in brackets represents the energy consumption at the forwarding node that occupies the medium first, and similarly the second term is the other neighbor nodes, which initially try to participate in local retransmission but suppress its packets after detecting an ongoing transmission attempt. While the latter do not transmit packets, they consume energy from random backoff and channel sensing until packet forwarding is detected.

Since the gateway exploits no backoff between repetitive transmissions, the mean idle listening period at each node becomes $t_{packet}/2$. If a destination receives a packet whose time index s is larger than 0, it sleeps again until the gateway transmits the packet with $s = 0$. The total energy consumption for a packet reception at the destination becomes

$$\begin{aligned} E^{rx}_{SHDP} &= \frac{3p_{rx}t_{packet}}{2} + (1-e)\,p_{rx}t_{packet} + (1-e)^2 p_{tx}t_{ack} \\ &\quad +(1-(1-e)^2)\left(1-(1-(1-e)^2)^{N_{ne}}\right)\varepsilon_{rx,one}. \end{aligned} \tag{33}$$

The first, second, and third terms come from direct packet receptions at the destination, and the last term results from the packet reception through neighbor forwarding.

In *SHDP*, all nodes overhear the gateway's transmission first, and then, only the destination's neighbors continue sensing to confirm packet delivery. Therefore, we obtain the total energy consumption for overhearing as

$$\begin{aligned} E^{oh}_{SHDP} &= \frac{3p_{rx}t_{packet}N_{tot}}{2} + (1-e)\,N_{ne}\,p_{rx}(t_{packet}+t_{ack}) \\ &\quad +(1-(1-e)^2)\left(1-(1-(1-e)^2)^{N_{ne}}\right)\varepsilon_{oh,one}. \end{aligned} \tag{34}$$

The first term here represents the energy consumption of all the nodes that overhear the gateway's transmission. The second term denotes the energy consumption of the

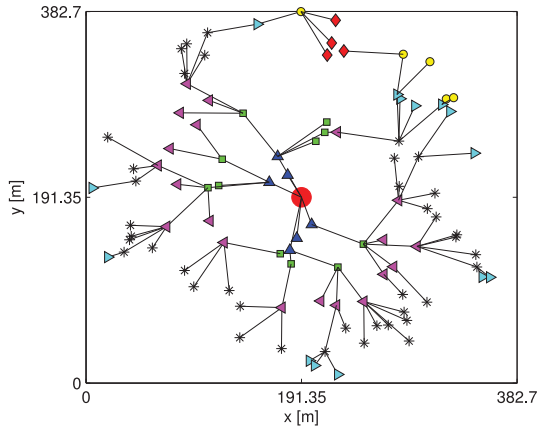


Fig. 5. Simulation network topology of 100 nodes.

destination's neighbors, which overhear the $s = 0$ packet and its ACK after successfully overhearing packet of index $s(>0)$. The last term is the energy consumption for overhearing the packet forwarded by a neighbor.

The total power consumption of all the nodes in *SHDP* can be written as

$$P_{SHDP} = p_{cs,tot} + \frac{E^{tx}_{SHDP} + E^{rx}_{SHDP} + E^{oh}_{SHDP}}{t_{int}}. \tag{35}$$

As in the analysis of MHDP energy consumption, P_{SHDP} is directly related with t_{wakeup} due to $p_{cs,tot}$ and $\varepsilon_{tx,one}$. The weight for $\varepsilon_{tx,one}$ in *SHDP* is lower than that of MHDP, since *SHDP* works independently from the number of nodes and its relaying occurs only when the gateway's transmission fails. The low transmission overhead allows *SHDP* to consume much lower energy with a large wakeup interval, especially in large-scale WSNs, compared to other multihop protocols.

5. SIMULATION RESULTS

Based on our mathematical analysis, we now move on to performance evaluation of *SHDP* and MHDP through simulation. Specifically, we developed an event-driven simulator that considers real channel environments according to the path loss model of Equation (6), which is suitable for indoor environments [Kim 2013]. Simulation results help us observe the performance of competitive schemes in terms of latency, packet delivery ratio, and power consumption for a large network size, which is not easy for experimental tests.

We configured a simulation environment similar to Kim et al. [2014]. We assume that each low-power node uses a transmission power of 0dBm (i.e., the maximum transmission power of CC2420), while the gateway in *SHDP* uses 17dBm as a WiFi transceiver [IEEE Std 802.11-2009 2009]. Our feasibility analysis presented in Section 2.1 shows that these transmission power settings allow each node to have transmission ranges of ∼58 meters and the gateway to have a range of ∼191 meters. As we show in Figure 5, we randomly deploy 100 nodes in a circular area with a radius of 191 meters and locate the gateway at the center, which is depicted as a red dotted circle. Considering that we apply the offset quadrature phase-shift keying (O-QPSK)

modulation of IEEE 802.15.4 [IEEE Std 802.15.4-2003 2003], we obtain the BER as

$$BER(\gamma) = \frac{8}{15} \times \frac{1}{16} \times \sum_{k-2}^{16} (-1)^k \binom{16}{k} \exp\left(20\left(\frac{1}{k} - 1\right)\gamma\right). \tag{36}$$

We set $p_{tx} = 70\text{mW}$,[4] $p_{rx} = 78\text{mW}$, $p_{cs} = 30\text{mW}$, and $p_{idle} = 3.7\text{mW}$ following the CC2420 specification [Texas Instruments 2006].

We assume that the channel-sensing range is the same as the transmission range, and do not consider the capture effect. As a result, a hidden-node collision always results in a packet reception error. Based on an extensive set of experiments, we configure each node to have a FIFO buffer size of 10 packets and a maximum of 10 transmission attempts for each packet delivery (i.e., $n_{tx} = 10$). Since we target downward traffic, the gateway generates a packet toward a randomly chosen destination every 15 seconds. Each simulation runs for 4 hours.

In Figure 5, the same mark represents the same hop distance from the gateway. Assuming that each node in MHDP knows its neighbor nodes and their depth from the gateway, it selects a neighbor with the minimum depth as its parent node. The gateway has the topology information and an ideal routing path to each destination. Each node is assumed to have routing information for each destination in advance without incurring any control overhead.

5.1. Latency

Figure 6 plots the average packet delivery latency observed by varying the wakeup interval. Here, the horizontally dotted red line indicates the interpacket interval (IPI) of 15 seconds. As expected, infrequent wakeup naturally results in increased delivery latency. *SHDP* achieves significantly lower latency than MHDP as the wakeup interval increases. This is mainly due to the fact that in *SHDP*, the higher transmission power of the gateway leads to direct one-hop delivery most of the time, whereas the same packet delivery requires multiple-hop transmission in MHDP.

Furthermore, we notice that the performance gap between analysis and simulation is larger with the wakeup interval in MHDP (especially when the latency becomes larger than IPI). This is because our analysis has not incorporated contention and collision effects on performance.

5.2. Packet Delivery Ratio

The packet delivery ratio is an important metric in many applications, which is shown in Figure 7. Although the analytical results show that MHDP is on par with *SHDP* in packet delivery performance, our simulation results differ a bit from those of analysis. Deep investigation on the simulation traces reveals that packet loss in MHDP is higher than that in *SHDP* when the wakeup interval is large, mainly due to the queueing loss. The queueing delay in MHDP results in high packet latency as presented in Figure 6. Therefore, buffer provisioning in MHDP is an important concern that *SHDP* avoids by transmitting downward packets in one hop.

5.3. Power Consumption

In Figure 8, we show the average power consumption per node with respect to the wakeup interval. The power consumption of MHDP first decreases and then increases with the wakeup interval. This is caused by the increased transmission burden from the wakeup interval. In contrast, *SHDP* consumes less power than MHDP, and its power consumption continuously decreases with the wakeup interval.

[4] p_{tx} is the *total* power consumption of a node in transmit mode.

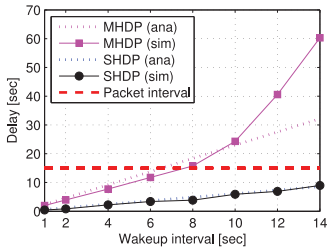


Fig. 6. Latency versus wakeup interval.

	2s	6s	10s	14s
MHDP (ana.)	10^{-5}	10^{-5}	10^{-5}	10^{-5}
MHDP (sim.)	0	0	0	$3 \cdot 10^{-2}$
SHDP (ana.)	10^{-5}	10^{-5}	10^{-5}	10^{-5}
SHDP (sim.)	0	0	0	0

Fig. 7. Packet delivery failure ratio versus wakeup interval.

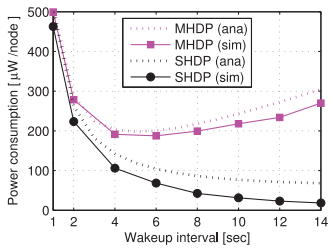


Fig. 8. Power consumption versus wakeup interval.

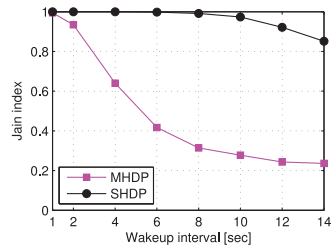


Fig. 9. Fairness in power consumption versus wakeup interval.

We break down the power consumption factors in detail using Figure 10. As expected, power consumption for channel sensing in MHDP and *SHDP* decreases with the wakeup interval. Regarding other power consumption factors, the transmission power consumption in MHDP increases with the wakeup interval due to prolonged repetitive transmission. On the other hand, since *SHDP* finishes its transmission mostly via single hop (from the gateway), the power consumption of the low-power nodes remains very low.

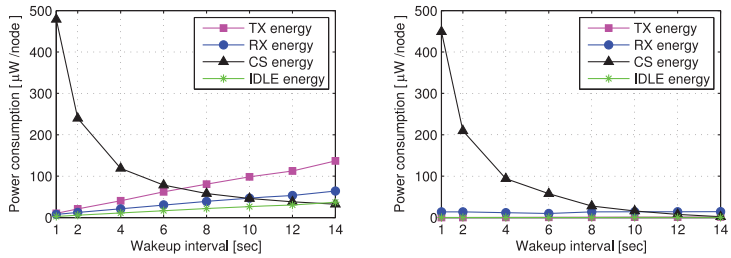
Lastly, we present the fairness in power consumption using the Jain's Fairness Index [Kim et al. 2014]:

$$J(p) = \frac{\left(\sum_{i=1}^{N_{tot}} p_i\right)^2}{N_{tot} \sum_{i=1}^{N_{tot}} p_i^2}, \tag{37}$$

where p_i is the power consumption of node i. In Figure 9, *SHDP* shows an overall fair performance compared to MHDP. This is mainly because in a multihop network, the routing burden on nodes that are closer to the gateway is unavoidably higher than that of the other nodes distant from the gateway [Kim et al. 2015b]. In *SHDP*, fair power consumption can be achieved since the gateway covers the whole network directly.

6. TESTBED EXPERIMENTS

We implemented SHDP on top of RPL and BoX-MAC-2 in TinyOS [Winter et al. 2012; Moss and Levis 2008]. RPL is an IP routing protocol for low-power and lossy networks that is widely used [Ko et al. 2010; Ancillotti et al. 2013]. Unlike the ideal (and unrealistic) MHDP, RPL requires control overhead in finding routes and suffers from limited or outdated routing information. Figure 11 summarizes some of the major modifications made to the software stack of TinyOS. Specifically, we removed the downlink route



(a) MHDP (b) SHDP

Fig. 10. Detailed power consumption breakdown.

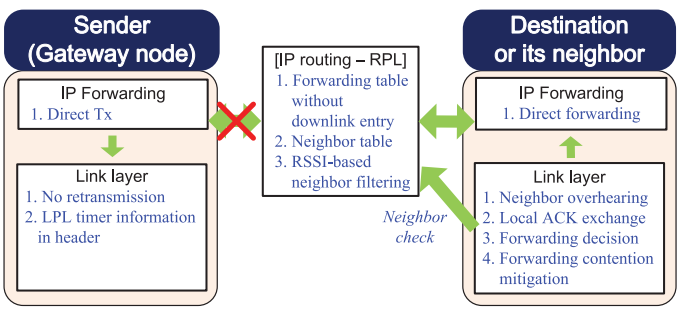


Fig. 11. SHDP architecture.

Fig. 12. Left: MTM-CM3300MSP for the high-power gateway. Right: Kmote for the low-power nodes.

discovery procedures and downward routing table in the IP-support stack. This allows downward packets to be sent directly to the destination within the software stack.

Furthermore, we swap the parent table in RPL with a *neighbor* table, since SHDP uses neighbor information for local ACK exchange and neighbor forwarding. If a node with an inconsistent link is added to the neighbor table, SHDP experiences a large neighbor-forwarding overhead since local ACK exchange becomes difficult. For efficient neighbor forwarding, each node adds a node to the neighbor table only when its observed RSSI is higher than a predefined threshold $p_{r,th}$. The upward route loss problem, which is potentially caused by the RSSI-based neighbor filtering in a low-density network, can be mitigated by activating the filtering process only when a node has a proper parent node already. SHDP adopts a cross-layer design approach in the sense that the link layer uses the routing layer information to confirm whether the received packet's destination is one of its neighbors.

Figure 13 shows the testbed topology map consisting of 20 nodes and one gateway, marked with a star, in an office environment (c.f., Figure 12 presents the hardware platforms used in this experiment). For a low-power node, we use the TelosB-clone device [Moteiv Corporation 2006], which combines an MSP430 microcontroller [Texas Instruments 2006] with a TI CC2420 transceiver [Texas Instruments 2006]. For the high-powered gateway, we use MTM-CM3300MSP [MAXFOR Technology 2014], which is similar to TelosB [Moteiv Corporation 2006] but includes a 10dB power amplifier. In our experiments, the high-power gateway and common low-power nodes use transmission power 10dBm and -15dBm, respectively, with an antenna gain of 5dB. To maintain both multihop uplink routes and single-hop downlink connectivity, the high-power gateway transmits a data packet with 10dBm and other packets (e.g., routing packets and ACKs) with -15dBm. Given this testbed configuration, the RPL

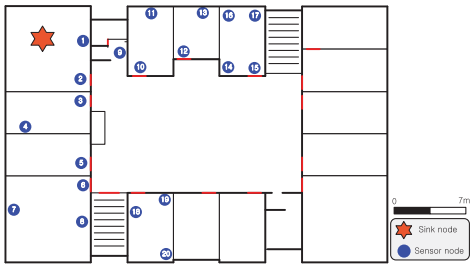


Fig. 13. Testbed topology map.

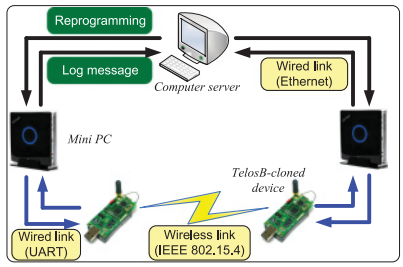
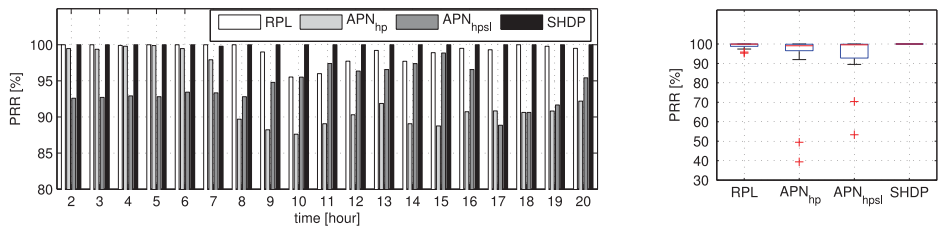


Fig. 14. Testbed architecture.



(a) Per-hour PRR performance (b) Per-node PRR performance

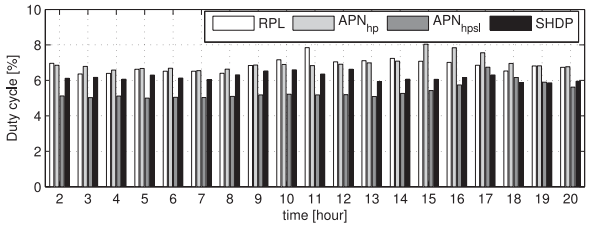
Fig. 15. PRR performance of RPL, APN$_{hp}$, APN$_{hpsl}$, and *SHDP* for 19 hours (from 2AM to 9PM), where APN$_{hp}$ and APN$_{hpsl}$ are variants of *SHDP*.

implementation connects all the nodes in a maximum of three hops. From empirical results, we set the threshold $p_{r,th}$ as -87dBm when deciding to include a node in the neighbor table. Finally, each node has a FIFO buffer size of 10 packets.

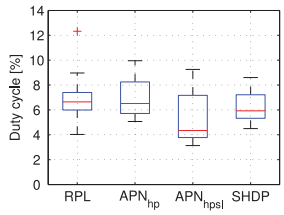
Finally, as we illustrate in Figure 14, each WSN node is connected to a PC via USB and sends log messages to the PC through the UART back-channel. We gather the log messages from each PC through the ethernet to obtain various performance measurements and real-time operation statuses. Furthermore, we remotely reprogram each node through the UART and ethernet back-channels. The two connections are only used for debugging and statistics gathering, and are not used for data communication between nodes. With this testbed architecture, we obtain various performance metrics by allowing each node to calculate its routing overhead, up/downward transmission overhead, and duty-cycle performance to be included in its log messages. Furthermore, we piggyback routing path information such as hop distance and end-to-end retransmissions in the application payload of each data packet and configure each relay node to update the information.

6.1. Downward Traffic Scenario

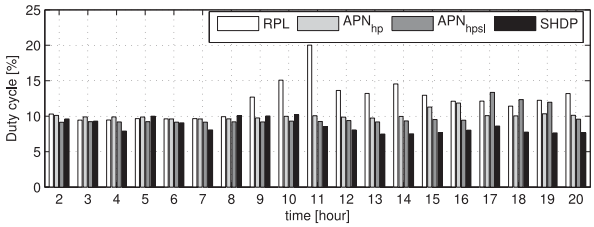
As a first step in our testbed evaluation, we consider a downward traffic scenario. We configure the gateway to generate packets toward each destination at an interval of 75 seconds (i.e., 3.75 seconds from the network's perspective) and set the asynchronous wakeup interval of each node to 0.5 second. To verify the effectiveness of each design element in *SHDP*, we also evaluate the performance of two variants of *SHDP*, termed as APN$_{hp}$ and APN$_{hpsl}$. APN$_{hp}$ includes the gateway's high-power transmission but does not allow each node to participate in neighbor forwarding nor sleep after overhearing the gateway's transmissions with $s > 0$. On the other hand, APN$_{hpsl}$ includes the gateway's high-power transmission and sleep-after-overhearing mechanism but does not use neighbor forwarding of *SHDP*.



(a) Average per-hour duty-cycle performance



(b) Per-node duty-cycle performance



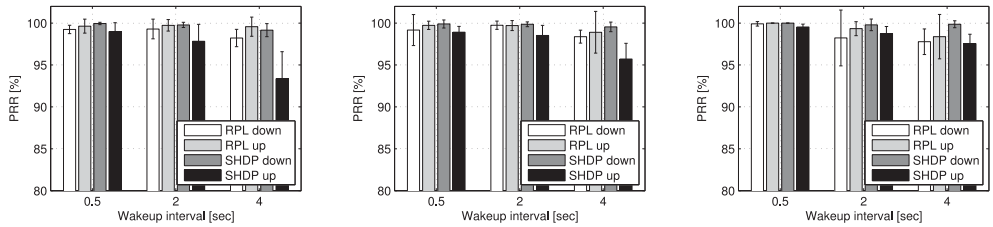
(c) Per-hour duty-cycle performance for the worst-performing node

Fig. 16. Duty-cycle performance of RPL, APN$_{hp}$, APN$_{hpsl}$, and *SHDP* for 19 hours (from 2AM to 9PM), where APN$_{hp}$ and APN$_{hpsl}$ are variants of *SHDP*.

In Figure 15(a), we present the per-hour average PRR throughout per day. During the daytime, the PRR performance of RPL fluctuates due to the channel dynamics created by the movement of people. We can observe that APN$_{hp}$ and APN$_{hpsl}$ are more vulnerable to channel dynamics, and their PRR performances are fluctuating and poorer compared to RPL. This reveals that only using a high-transmission-power gateway is not sufficient for achieving reliable downward packet delivery in APNs. On the other hand, in *SHDP*, we always observe not only the most stable but also the highest PRR performance among the competitive protocols mainly owing to the combination of high-power single-hop transmissions at the gateway and local retransmissions at neighbor nodes of the packet's destination. This implies that the link instability due to channel dynamics significantly impacts the performance of both multihop and high-powered single-hop communications, while this is not an issue in *SHDP*.

Figure 15(b) plots the PRR performance of each node for the four protocols of our interest. The results show that APN$_{hp}$ and APN$_{hpsl}$ significantly suffer from unfair PRR performance among the nodes. By matching the results with the physical topology depicted in Figure 13, we confirm that nodes experiencing very low PRR are far from the gateway (e.g., nodes 18–20) or hidden behind obstacles (e.g., node 11). This reveals that, in practice, high-powered single-hop transmission is difficult to guarantee reliability for nodes placed at the boundary of the transmission range. Lastly, *SHDP* provides better, more stable, and fairer PRR performance than the others by overcoming both path loss and channel dynamics, which verifies that neighbor forwarding in *SHDP* is beneficial to maximizing the strength of APNs and leads to reliable packet delivery.

We further focus on the duty-cycle performance in Figures 16(a) through 16(c), which indirectly represents the nodes' energy consumption. First, Figure 16(b) shows that RPL's duty-cycle performance is significantly unfair among nodes. The unbalanced transmission overhead among nodes in RPL is unavoidable due to its multihop nature since nodes near the gateway are asked to relay more packets compared to close-to-leaf nodes. Furthermore, RPL inherits the load-balancing problem [Kim et al. 2015b],



(a) Downlink interval = 100s, uplink interval = 300s (b) Downlink interval = 150s, uplink interval = 150s (c) Downlink interval = 300s, uplink interval = 100s

Fig. 17. Average PRR performance versus wakeup interval.

which worsens the unfair energy consumption among nodes. From Figures 16(a) and 16(c), we can observe that the duty-cycle performance of RPL fluctuates in the time domain, which becomes severe for the worst-performing node. This is because RPL requires a high number of retransmissions and control overhead for achieving reliable packet delivery and route maintenance with fluctuating links. Despite the effort, RPL fails to overcome link instability as presented in Figure 15(a).

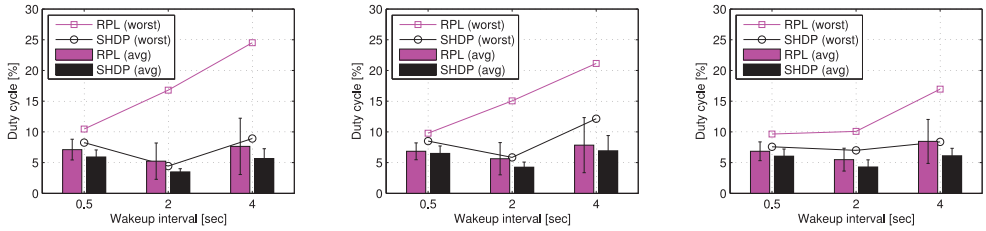
Interestingly, the results here reveal that APN_{hp} consumes more energy than RPL since each high-powered transmission triggers idle listening at all nodes, leading to a waste of energy. On the other hand, APN_{hpsl} provides the lowest duty-cycle performance by allowing each node to sleep after overhearing the gateway's packet transmission with $s > 0$ without idle listening. Lastly, we can see that the average per-hour duty-cycle performance of *SHDP* is slightly lower than that of RPL, and for the worst case, the differences become far more prominent. This is because *SHDP* provides low and fair transmission overhead for nodes by allowing the gateway to take most of the transmission burden. Quantitatively, *SHDP*'s average transmission overhead is only one-third of RPL's. For the worst-case node, this gap increases by $\frac{1}{18}$. Compared to APN_{hpsl}, *SHDP* provides only slightly higher radio duty cycles due to the neighbor-forwarding overhead. Given other performance metrics, we find this as a reasonable cost to pay.

From the experimental results in this section, we can confirm that, while we can easily obtain the APN architecture in terms of hardware configurations, its performance heavily depends on how the network protocol is designed. *SHDP* successfully takes the advantages of an APN architecture and significantly improves the networking performance over RPL.

6.2. Mixed Traffic Scenario

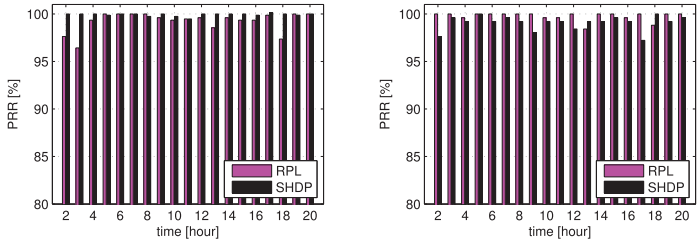
In most cases, even if downward traffic takes a major portion of network traffic, some upward traffic coexists. For this, we vary the upward traffic generation interval at each node from 100 seconds to 300 seconds and adjust the downward traffic generation interval to vary between 100 seconds and 300 seconds. We test three cases of downward and upward traffic generation intervals of [100 seconds, 300 seconds], [150 seconds, 150 seconds], and [300 seconds, 100 seconds], respectively. We consider wakeup intervals of 0.5, 2, and 4 seconds.

In Figure 17, we plot the PRRs for the three offered traffic cases. While the performance details are different in each case, we notice that the downward packet delivery performance in *SHDP* is superior to that of RPL in all cases. For upward traffic, *SHDP* is on par with RPL for short wakeup intervals and shows slightly decreased performance with the wakeup interval. This is mainly because high-power downward broadcast packets can be queued due to infrequent wakeups, and this congestion causes packet loss at low-power nodes. *SHDP* provides nearly perfect downward PRR regardless



(a) Downlink interval = 100s, uplink interval = 300s (b) Downlink interval = 150s, uplink interval = 150s (c) Downlink interval = 300s, uplink interval = 100s

Fig. 18. Duty-cycle performance versus wakeup interval.



(a) Downward PRR (b) Upward PRR

Fig. 19. Per-hour average PRR performance for downward and upward traffic.

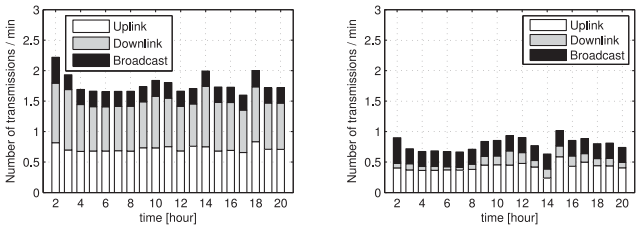
of traffic patterns, but its upward PRR decreases as the downward traffic becomes dominant. Each high-powered transmission in *SHDP* incurs wireless interference throughout the whole network, which causes frequent packet collisions at low-power nodes when the network generates heavy downward traffic.[5]

In Figure 18, we note an interesting observation for the duty cycle of each node in both the average case and the worst case. For all the nine instances, the average duty cycle of *SHDP* is lower than that of RPL. The duty cycle observed from the worst-case node in *SHDP* is significantly lower compared to that of RPL. This implies that *SHDP* has a significantly longer network lifetime than RPL since the battery lifetime of the first dead node impacts the usability of the entire system [Ganesh and Amutha 2013].

Next we turn our attention to the per-hour performance of the test network. For this, we take the traffic case of [100 seconds, 300 seconds] with the wakeup interval of 0.5 second and plot the PRR performance for each hour in Figure 19. In Figure 19(a), *SHDP* shows steady downward performance over time. This result implies that the high-power transmission and local retransmission features of *SHDP* help to achieve steady packet delivery performance despite the varying channel condition throughout the day. For upward PRR performance shown in Figure 19(b), RPL shows slightly higher PRR than *SHDP*. This is because RPL is optimized for data collection, and downward routes are simply set as the reverse of upward routes.

Figure 20 shows the overhead for each packet transmission. The overhead in *SHDP* is significantly lower than that in RPL for both downward and upward transmissions. The main reason for the reduced upward transmission overhead is that RPL forces each node to transmit destination advertise object (DAO) packets to the gateway as a way of maintaining downward routes, while *SHDP* does not generate such control packets.

[5]We consider improving upward packet delivery performance in APNs as part of our future work.



(a) RPL transmission overhead (b) *SHDP* transmission overhead

Fig. 20. Per-hour average transmission overhead.

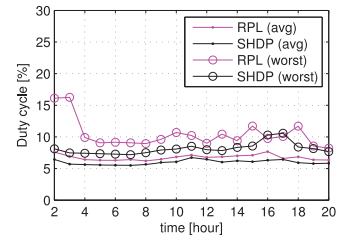


Fig. 21. Per-hour duty-cycle performance.

Lastly, we observe the average per-hour duty cycle of each node and that of the worst-case node over time in Figure 21. When combining the results of Figure 16 and Figure 21, we observe that each node in *SHDP* has a lower duty cycle compared to RPL under various traffic scenarios. Overall, our experimental results show that in a real-world test environment, *SHDP* outperforms the multihop routing protocol of RPL when reliable and energy-efficient delivery of downward packets is considered.

7. RELATED WORK

Ko et al. [2011] experimentally evaluated the performance of RPL using the TinyRPL implementation in TinyOS and have shown that its performance is similar to that of the collection tree protocol (CTP), the de facto data collection protocol in TinyOS, while having the benefits of an IPv6-based architecture. Kim et al. [2015a] evaluated the performance of TCP over RPL on a multihop WSN testbed. These works revealed that RPL's downward packet delivery performance is less efficient than the upward performance.

A number of works revealed issues related to RPL and tried to alleviate these problems. Ancillotti et al. [2014] proposed a cross-layering design for RPL, which provides enhanced link estimation and efficient management of neighbor tables. They used AMI as a case study and employed the Cooja emulator to evaluate their proposal. The work in Kim et al. [2015b] investigated the load-balancing problem of RPL and revealed that the performance of RPL severely degrades in a heavy traffic environment due to queue loss. To alleviate the issue, QU-RPL was proposed to allow each node to smartly use queue utilization information for its routing parent selection. Even though these works provided better upward routes, they did not consider the downward traffic delivery performance of RPL.

In the perspective of downward route management, Ko et al. [2015] showed that RPL has a serious connectivity problem when two modes of operation (MOPs) are mixed within a single network. To address this issue, the authors proposed DualMOP-RPL, which supports nodes with different MOPs to communicate gracefully in a single network while preserving the high bidirectional data delivery performance. However, this work focused on addressing interoperability problems between two MOPs rather than finding better downward routes.

There are several deployment examples of multihop WSNs. Ko et al. [2010] deployed a CTP-based WSN at Johns Hopkins Hospital and showed that multihop WSNs can provide reasonable performance for monitoring the status of patients. Cisco [2015] designed and deployed a field area network (FAN) for smart grids (CG-Mesh), which uses 6LoWPAN [Montenegro et al. 2007], RPL, and IPv6 on top of IEEE 802.15.4 to provide end-to-end two-way communication to each smart metering endpoint. Buevich et al. [2014] presented the architecture, design, and experiences from a wirelessly managed microgrid deployment in a rural area. They constructed a multihop network

by using their own flooding mechanism mPCF. However, none of these works consider downward-centric traffic patterns. Most importantly, all of the aforementioned works assume that all nodes use the same transmission power, which is different from our work.

Several works considered asymmetric (heterogeneous) transmission power usage among nodes. Li and Hou [2004] proposed a topology control algorithm when each node exploits different transmission powers. Lin et al. [2006] designed each node to control its transmission power considering wireless network connectivity. Although these works considered link asymmetry among nodes, they still constructed homogeneous networks in the routing perspective. That is, they constructed multihop networks in which bidirectional routes have the same hop distance, instead of creating multihop upward and single-hop downward connectivity. Li and Fang [2012] analyzed throughput performance in a heterogeneous wireless network such as a city-wide wireless network and battlefield network, where some powerful nodes are deployed along with normal nodes. This work provided a theoretical approach with high algorithm complexity, thereby leaving some distance from being implemented on real devices.

8. CONCLUSION

In this article, we present the SHDP protocol, which exploits asymmetric transmission power capabilities of heterogeneous nodes. Specifically, in our network environment, the gateway node uses a strong transmission power to cover the entire network in a single hop while common sensor nodes maintain low-transmission-power profiles. SHDP comprises subcomponents such as the gateway's high-power transmission, packet overhearing by neighbor nodes of a destination, local ACK exchange between a destination and its neighbors, neighbor forwarding in the case of direct transmission failure from the gateway, and contention mitigating between neighbors.

Through mathematical analysis, extensive simulation, and empirical testbed experiments, we confirm that SHDP outperforms other competitive multihop routing protocols with respect to PRR performance, radio duty cycle, and transmission overhead. We see that our work on asymmetric transmission power-based nodes will bring new perspectives in designing sensing systems for various application domains: with low-cost radio modules and reduction in hardware development costs, the assumption of ubiquitously using homogeneous transmission power-based networks may not be the most efficient design choice anymore.

REFERENCES

E. Ancillotti, R. Bruno, and M. Conti. 2013. The role of the RPL routing protocol for smart grid communications. *IEEE Communications Magazine* 51, 1 (Jan. 2013), 75–83. DOI:http://dx.doi.org/10.1109/MCOM.2013.6400442

E. Ancillotti, R. Bruno, and M. Conti. 2014. Reliable data delivery with the IETF routing protocol for low-power and lossy networks. *IEEE Transactions on Industrial Informatics* 10, 3 (Aug. 2014), 1864–1877. DOI:http://dx.doi.org/10.1109/TII.2014.2332117

Aruba Networks. 2012. Technology advances in retail: Improving margins using wireless networks. Available at: http://www.arubanetworks.com/pdf/technology/whitepapers/wp_Retail_advances.pdf. (2012).

M. Buevich, D. Schnitzer, T. Escalada, A. Jacquiau-Chamski, and A. Rowe. 2014. Fine-grained remote monitoring, control and pre-paid electrical service in rural microgrids. In *Proceedings of the 13th International Symposium on Information Processing in Sensor Networks (IPSN'14)*. 1–11. DOI:http://dx.doi.org/10.1109/IPSN.2014.6846736

M. Ceriotti, M. Corra, L. D'Orazio, R. Doriguzzi, D. Facchin, S. T. Guna, G. P. Jesi, R. Lo Cigno, L. Mottola, A. L. Murphy, M. Pescalli, G. P. Picco, D. Pregnolato, and C. Torghele. 2011. Is there light at the ends of the tunnel? Wireless sensor networks for adaptive lighting in road tunnels. In *2011 10th International Conference on Information Processing in Sensor Networks (IPSN'11)*. 187–198.

Y. Chen and A. Terzis. 2011. On the implications of the log-normal path loss model: An efficient method to deploy and move sensor motes. In *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys'11)*. 26–39. DOI:http://dx.doi.org/10.1145/2070942.2070946

Y.-J. Choi, S. Park, and S. Bahk. 2006. Multichannel random access in OFDMA wireless networks. *IEEE Journal on Selected Areas in Communications* 24, 3 (Mar. 2006), 603–613. DOI:http://dx.doi.org/10.1109/JSAC.2005.862422

Cisco. 2015. Connected grid networks for smart grid - field area network. http://www.cisco.com/web/strategy/energy/field_area_network.html. (2015).

T. Clausen, U. Herberg, and M. Philipp. 2011. A critical evaluation of the IPv6 routing protocol for low power and lossy networks (RPL). In *2011 IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob'11)*. 365–372. DOI:http://dx.doi.org/10.1109/WiMOB.2011.6085374

Federal Communications Commission. 1993. FCC power regulation. Available at: http://transition.fcc.gov/Bureaus/Engineering Technology/Documents/bulletins/oet63/oet63rev.pdf. (1993).

J. Fink, A. Ribeiro, and V. Kumar. 2012. Robust control for mobility and wireless communication in cyber physical systems with application to robot teams. *Proceedings of the IEEE* 100, 1 (Jan. 2012), 164–178. DOI:http://dx.doi.org/10.1109/JPROC.2011.2161427

S. Ganesh and R. Amutha. 2013. Efficient and secure routing protocol for wireless sensor networks through SNR based dynamic clustering mechanisms. *Journal of Communications and Networks* 15, 4 (Aug. 2013), 422–429. DOI:http://dx.doi.org/10.1109/JCN.2013.000073

O. Gnawali, R. Fonseca, K. Jamieson, and P. Levis. 2008. CTP: Robust and efficient collection through control and data plane integration. Stanford Information Networks Group Technical Report SING-08-02. (2008).

IEEE Std 802.11-2009 2009. IEEE Standard for Information Technology– Local and Metropolitan Area Networks– Specific Requirements– Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput. (Oct 2009). DOI:http://dx.doi.org/10.1109/IEEESTD.2009.5307322

IEEE Std 802.15.4-2003 2003. Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). Available at http://www.ieee802.org/15/pub/TG4.html. (May 2003).

H.-S. Kim. 2013. SHDP simulator. Available at: http://netlab.snu.ac.kr/∼hskim. (2013).

H.-S. Kim, Y.-J. Choi, and S. Bahk. 2014. Elimination of multi-hop transmission form downlink in low power and lossy networks. In *Proceedings of IEEE Conference on Communications (ICC'14)*.

H.-S. Kim, H. Im, M.-S. Lee, J. Paek, and S. Bahk. 2015a. A measurement study of TCP over RPL in low-power and lossy networks. *Journal of Communications and Networks* 17, 6 (Dec. 2015), 647–655.

H.-S. Kim, J. Paek, and S. Bahk. 2015b. QU-RPL: Queue utilization based RPL for load balancing in large scale industrial applications. In *IEEE International Conference on Sensing, Communication, and Networking (SECON'15)*.

J.-G. Ko, S. Dawson-Haggerty, O. Gnawali, D. Culler, and A. Terzis. 2011. Evaluating the performance of RPL and 6LoWPAN in TinyOS. In *Workshop on Extending the Internet to Low Power and Lossy Networks (IP+SN'11)*.

J. Ko, J. Jeong, J. Park, J. A. Jun, O. Gnawali, and J. Paek. 2015. DualMOP-RPL: Supporting multiple modes of downward routing in a single RPL network. *ACM Transactions on Sensor Networks* 11, 2, Article 39 (2015), 20 pages. DOI:http://dx.doi.org/10.1145/2700261

J.-G. Ko, J.-H. Lim, Y. Chen, R. Musaloiu-E., A. Terzis, G. Masson, T. Gao, W. Destler, L. Selavo, and R. Dutton. 2010. MEDiSN: Medical emergency detection in sensor networks. *ACM Transactions on Embedded Computing Systems (TECS), Special Issue on Wireless Health Systems* (2010).

J.-G. Ko, A. Terzis, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, and P. Levis. 2011. Connecting low-power and lossy networks to the Internet. *IEEE Communications Magazine* 49, 4 (Apr. 2011), 96–101. DOI:http://dx.doi.org/10.1109/MCOM.2011.5741163

O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson. 2012. Low power, low delay: Opportunistic routing meets duty cycling. In *Proceedings of International Conference on Information Processing in Sensor Networks (IPSN'12)*. 185–196. DOI:http://dx.doi.org/10.1145/2185677.2185731

N. Li and J. C. Hou. 2004. Topology control in heterogeneous wireless networks: problems and solutions. In *23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, Vol. 1. 243. DOI:http://dx.doi.org/10.1109/INFCOM.2004.1354497

P. Li and Y. Fang. 2012. On the throughput capacity of heterogeneous wireless networks. *IEEE Transactions on Mobile Computing* 11, 12 (Dec. 2012), 2073–2086. DOI:http://dx.doi.org/10.1109/TMC.2011.239

S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He. 2006. ATPC: Adaptive transmission power control for wireless sensor networks. In *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys'06)*. ACM, New York, NY, 223–236. DOI:http://dx.doi.org/10.1145/1182807.1182830

K. Lorincz, D. J. Malan, T. R. F. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, and S. Moulton. 2004. Sensor networks for emergency response: challenges and opportunities. *IEEE Pervasive Computing* 3, 4 (Oct. 2004), 16–23. DOI:http://dx.doi.org/10.1109/MPRV.2004.18

M. Marot, B. Kusy, G. Simon, and A. Ledeczi. 2004. The flooding time synchronization protocol. In *Proceedings of the 2nd Conference on Embedded Networked Sensor Systems (SenSys'04)*. 39–49.

MAXFOR Technology. 2014. WSN communication module, MTM-CM3300MSP. Available at: http://www.maxfor.co.kr/datasheet/MAXFOR_Digital_Brochure.pdf. (2014).

G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler. 2007. Transmission of IPv6 packets over IEEE 802.15.4 networks. *RFC 4944* (Sept. 2007).

D. Moss and P. Levis. 2008. *BoX-MACs: Exploiting Physical and Link Layer Boundaries in Low-Power Networking*. Technical Report SING-08-00. Stanford Information Networks Group.

Moteiv Corporation. 2006. Tmote Sky Datasheet. http://www.moteiv.com/products/docs/tmote-sky-datasheet.pdf. (2006).

W. Pak, K.-T. Cho, J. Lee, and S. Bahk. 2008. W-MAC: Supporting ultra low duty cycle in wireless sensor networks. In *Proceedings of IEEE Global Communications Conference (GLOBECOM'08)*. 1–5. DOI:http://dx.doi.org/10.1109/GLOCOM.2008.ECP.79

D. Puccinelli, S. Giordano, M. Zuniga, and P. J. Marrón. 2012. Broadcast-free collection protocol. In *Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys'12)*. 29–42. DOI:http://dx.doi.org/10.1145/2426656.2426660

K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis. 2010. An empirical study of low-power wireless. *ACM Transactions on Sensor Networking* 6, 2, Article 16 (March 2010), 49 pages. DOI:http://dx.doi.org/10.1145/1689239.1689246

Texas Instruments. 2006. 2.4 GHz IEEE 802.15.4/ZigBee-ready RF transceiver. Available at: http://focus.ti.com/lit/ds/symlink/cc2420.pdf. (2006).

Texas Instruments. 2006. MSP430 Ultra-low-power microcontrollers. (2006).

F. Wang and J. Liu. 2009. Duty-cycle-aware broadcast in wireless sensor networks. In *IEEE Conference on Computer Communications (INFOCOM'09)*. 468–476. DOI:http://dx.doi.org/10.1109/INFCOM.2009.5061952

G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh. 2006. Fidelity and yield in a volcano monitoring sensor network. In *Proceedings of the 7th Symposium on Operating Systems Design and Implementation (OSDI'06)*. USENIX Association, Berkeley, CA, 381–396. http://dl.acm.org/citation.cfm?id=1298455.1298491

T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. P. Vasseur, and R. Alexander. 2012. RPL: IPv6 routing protocol for low-power and lossy networks. *RFC 6550* (2012).

K. S. Yildirim and A. Kantarci. 2014. Time synchronization based on slow-flooding in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems* 25, 1 (Jan. 2014), 244–253. DOI:http://dx.doi.org/10.1109/TPDS.2013.40

Received October 2014; revised October 2015; accepted August 2016