



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학박사 학위논문

An Asymmetric Transmission
Power-based Wireless Sensor System

비대칭 전송전력 기반 무선 센서 시스템 연구

2016년 2월

서울대학교 대학원

전기·컴퓨터 공학부

김 형 신

Abstract

Updating price tags in a large-scale market is a recurrent task, still performed manually in most markets. Given that human-errors can easily lead to customer complaints and accounting inaccuracies, the ability to autonomously reconfigure price tags can be of significant benefit. With the introduction of low-power display techniques such as electronic-ink, applications of enabling electronic, wirelessly reconfigurable price tags show potential for future deployment. In this dissertation, we examine networking architectures that can be applied in such scenarios. Through a series of preliminary pilot studies in an actual supermarket, we show that the performance of existing protocols are *not* ready to overcome the unique challenges of busy market environments. We identify underlying technical challenges and propose *MarketNet*, an asymmetric transmission power-based system designed for densely populated, obstacle-rich, downwards traffic-oriented environments. We evaluate *MarketNet* in a large indoor market visited by 5000+ customers per day. Our results show that *MarketNet* addresses the challenges of the target application and environment, while

achieving higher packet delivery performance with noticeably lower radio duty-cycles than existing protocols such as RPL and LPL.

Keywords: price tag, Internet of Things, routing protocol, wireless sensor network, low power and lossy network, IPv6

Student Number: 2011-30227

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Related Work	4
1.2.1	Wireless Price Tag Update System	4
1.2.2	Wireless Systems Community	5
1.2.3	Wireless Network Community	10
1.3	Contributions and Outline	12
2	Target Application: Wireless and Remote Update of e-Price Tags	17
2.1	Price Representation	17
2.2	Application Scenario	19
2.3	System Requirements	22
3	Preliminary Study in Urban Crowded Markets	25
3.1	Introduction	25
3.2	Wireless Channel Characteristics	27

3.3	Performance of Pre-existing Protocols	38
3.3.1	RPL Operation	38
3.3.2	LPL Operation	42
3.3.3	Performance of RPL over LPL	44
3.4	Summary	48
4	MarketNet 1.0: Asymmetric Transmission Power-based Network	50
4.1	Introduction	50
4.2	Applicability of Asymmetric Transmission Power Networks	54
4.3	<i>MarketNet1.0</i> System Design	58
4.3.1	Design Overview	58
4.3.2	Neighbor Forwarding over LPL	61
4.4	Mathematical Performance Analysis	66
4.4.1	Packet Reception Ratio	67
4.4.2	Latency	69
4.4.3	Power Consumption	70
4.5	Simulation Results	78
4.5.1	Latency	80
4.5.2	Packet Delivery Ratio	81
4.5.3	Power Consumption	82
4.6	Testbed Experiments	84
4.6.1	Implementation and Environment Setting	84

4.6.2	Downward Traffic Scenario	88
4.6.3	Mixed Traffic Scenario	93
4.7	Market Deployment	98
4.8	Summary	102
5	MarketNet 2.0: Network-wide Superframe Architec- ture	103
5.1	Introduction	103
5.2	MarketNet2.0 System Design	105
5.2.1	Network-wide Superframe Architecture	107
5.2.2	IPv6 and Routing Layers in MarketNet2.0	112
5.3	System Evaluation	115
5.3.1	Testbed Evaluations	115
5.3.2	Market Deployments	120
5.4	Non-technical but Practical Lessons	126
5.5	Summary	127
6	Conclusion	129
6.1	Research Contributions	129
6.2	Further Research Direction	131

List of Tables

4.1	Packet delivery failure ratio vs. sleep interval	82
-----	--	----

List of Figures

1.1	An example of manual price tag updates by market staffs, which is a labor-intensive and error-prone task.	3
1.2	Our experimental market environment: a crowded in- door market with a size of 90×60 meters, >10k types of items, and >5000 customers per day.	13
2.1	Application scenario for our wirelessly reconfigurable e-price tagging system, consisting of a wireless-enabled central server and end devices such as price tags and shopping carts.	20
3.1	Various environments that have been explored by wire- less systems researchers. From the upperleft, island, wildlife, volcano, bird nest, woods, bridges, hospitals, and rural areas.	26
3.2	Link testing environment.	27
3.3	Low power node for measuring wireless link character- istics in markets.	28

3.4	Link characteristics of our indoor market environment for channel 26. Packet delivery performance degrades due to the effect of metal shelves and human activities during the day-time.	29
3.5	1 kHz RSSI sampling traces. The country we test in (South Korea) allows for WiFi traffic on channel 26, causing noise spikes during the day.	31
3.6	Characteristics of links 1 and 2 over time for channel 26. General human activities and some market-specific activities cause links to fluctuate on both short and long term perspectives.	33
3.7	An event of refilling items on a shelf, performed by a market staff. Some market-specific activities can change wireless environment and cause long term fluctuation on link quality.	34
3.8	1 kHz RSSI sampling on channel 26, performed near a microwave oven. Microwave oven is another major factor of link quality fluctuation.	35
3.9	Per hour conditional packet delivery function (CPDF) for different hours during the day, along with the negative tail (or left tail) lengths' correlation with human population. The negative tail length (i.e., positive burstiness) decreases with the number of customers. .	37

3.10	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 trade-off between a sender and a receiver in energy consumption according to the sleep interval.	42
3.11	Topology map of our 30 node market environment with a snapshot of RPL routes.	44
3.12	Packet delivery performance of RPL over LPL, both for uplink and downlink.	46
3.13	Duty cycle performance of RPL over LPL.	47
4.1	Enabling factor for an APN.	57
4.2	Design elements of <i>MarketNet1.0</i>	59
4.3	An example of direct transmission in <i>MarketNet1.0</i> . 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.	62

4.4	An example of neighbor forwarding in <i>MarketNet1.0</i> . A forwarder node can opportunistically retransmit an overheard downward packet to increase network reli- ability. To mitigate contention, the (potential) for- warder node suppresses forwarding when detecting an ongoing forwarding process of the same packet by an- other node.	65
4.5	Simulation network topology of 100 nodes.	79
4.6	Latency vs. sleep interval.	81
4.7	Power consumption vs. sleep interval.	83
4.8	Detailed power consumption breakdown.	83
4.9	Fairness in power consumption vs. sleep interval. . . .	84
4.10	<i>MarketNet1.0</i> architecture.	85
4.11	Testbed topology map.	86
4.12	Left: MTM-CM3300MSP for the high-power gateway, Right: Kmote for the low-power nodes.	86
4.13	Testbed architecture.	87
4.14	PRR performance of RPL, APN_{hp} , APN_{hpsl} , and <i>Mar-</i> <i>ketNet1.0</i> for 19 hours (from 2AM to 9PM), where APN_{hp} and APN_{hpsl} are variants of <i>MarketNet1.0</i> . . .	89
4.15	Duty cycle performance of RPL, APN_{hp} , APN_{hpsl} , and <i>MarketNet1.0</i> for 19 hours (from 2AM to 9PM), where APN_{hp} and APN_{hpsl} are variants of <i>MarketNet1.0</i> . . .	91
4.16	Average PRR performance vs. sleep interval.	94

4.17	Duty cycle performance vs. sleep interval.	95
4.18	Per-hour average PRR performance for downward and upward traffic.	96
4.19	Per-hour average transmission overhead.	97
4.20	Per-hour duty cycle performance.	97
4.21	Packet loss ratio of RPL and <i>MarketNet1.0</i> for uplink and downlink traffic. <i>MarketNet1.0</i> shows a lower loss rate for downlink traffic, while RPL outperforms on uplink packet delivery performance.	98
4.22	Transmission overhead of RPL and <i>MarketNet1.0</i> . <i>Mar-</i> <i>ketNet1.0</i> provides lower overhead by suppressing DAO messages and multihop forwarding.	99
4.23	Duty-cycle of RPL and <i>MarketNet1.0</i> . <i>MarketNet1.0</i> improves duty-cycle performance due to lower trans- mission overhead.	100
5.1	Design elements of <i>MarketNet2.0</i>	105
5.2	Superframe structure of <i>MarketNet2.0</i> . Downlink and uplink transmission periods are designated using a bea- con message sent by the root.	107
5.3	An example of <i>MarketNet2.0</i> 's initial synchronization procedure. The root broadcasts sync beacons in the inactive period for new (unsynchronized) nodes to join the superframe architecture.	111

5.4	DIO transmission interval vs. DIO transmission counter for RPL and <i>MarketNet2.0</i>	114
5.5	Topology map of indoor 30-node testbed with a snap- shot of RPL's routing paths.	115
5.6	Loss-rate and radio duty-cycle results from the testbed for varying sleep intervals and <i>MarketNet2.0</i> . <i>Market- Net2.0</i> provides the lowest networking overhead and duty-cycles.	117
5.7	Loss-rate and duty-cycle results from the testbed for varying uplink and downlink traffic interval patterns. <i>MarketNet2.0</i> shows the lowest packet loss and duty- cycles.	118
5.8	Impact of uplink partitioning. Uplink partitioning min- imizes both the per-link ETX and packet loss ratios for uplink traffic by separating transmission slots.	120
5.9	Network performance over 10-hour period for RPL, <i>Mar- ketNet1.0</i> and <i>MarketNet2.0</i> in the market environment. While the real market environment introduces an ad- ditional level of fluctuation over time, the performance trends of <i>MarketNet2.0</i> match our testbed results. . .	121
5.10	Downlink PRR of <i>MarketNet2.0</i> with and without neigh- bor forwarding scheme. Neighbor forwarding signifi- cantly improves reliability when transmission of high power root suffers from link dynamics or path loss. . .	123

5.11	Transmission overhead and duty-cycle fairness in the market for RPL, <i>MarketNet1.0</i> and <i>MarketNet2.0</i> . The reduction in transmission overhead is due to the suppression of DAO transmissions and packet forwarding for downwards delivery. Furthermore, <i>MarketNet2.0</i> shows a fair duty-cycle performance among the deployed nodes.	124
------	---	-----

Chapter 1

Introduction

1.1 Motivation

Low-power embedded wireless sensor networking technologies, with their ease of deployment and ubiquitous connectivity, have the potential to empower a number of real-world applications [1, 2, 3, 4, 5]. Many of these are designed around our everyday environments to simplify routine tasks that are straightforward, repetitive, and both time- and labor-intensive. Some applications are near commercialization, and a number of systems have been developed using various wireless technologies [6, 7].

On top of many useful applications, this dissertation focuses on making urban markets smarter. Markets are prevalent everywhere and take up a huge part of our everyday life. They provide a variety of products and we frequently go there to get those products for our

daily use. Thus, automating inefficient tasks in markets has potential to improve quality of our lives.

Among many routine tasks that occur in markets, this dissertation focuses on price tag reconfiguration, which is one of the most straightforward, repetitive, and both time- and labor- intensive tasks. Since the price has a significant effect on customers' behaviors, it changes frequently depending on their purchase pattern, inventory, competitive markets' behavior, and product freshness. For example, our collaborators and a number of previous work indicate that for items such as fresh grocery, on average, the prices change up to eight times a day. Furthermore, in typical markets, there are several types of prices such as membership, non-member, discount and buy one get one free (BOGO). Given a large number of items carried in large-sized markets, the management of these different types of time-varying prices is prone to errors and requires significant labor. However, it is still performed manually in most markets as in Figure 1.1, which not only incurs labor cost, but also increases customer complaints due to frequent human errors [6]. These problems motivate us to design an electronic price tagging system which remotely reconfigures electronic price tags (e-price tags) using low power wireless network.

With the development of low power display technologies such as electronic ink, e-price tag system is being an interesting application for low-power embedded systems. However, as a research community, we are still far from understanding the real wireless channel character-



Figure 1.1: An example of manual price tag updates by market staffs, which is a labor-intensive and error-prone task.

istics in such human-active and obstacle-present environments. Furthermore, while these e-price tags should be reconfigured wirelessly, we still lack knowledge on how these environments challenge existing wireless networking protocols. For example, it is unclear (practically) whether the use of multi-hop networking benefits or harms the application-level performance, or whether an alternate architecture such as asymmetric transmission power-based networks is a better fit in such scenarios. Thus, we investigate wireless link characteristics in an urban crowded marketplace and design a wireless network architecture *MarketNet* to support the e-price tagging application in the target environments.

1.2 Related Work

1.2.1 Wireless Price Tag Update System

Point-of-sale (POS) terminals, electronic cash registers, computers, bar-code scanners, and many other products are being used today to automate the retail business. However, one difficult-to-automate process is ensuring that the prices shown on store shelves agree with those displayed and registered at the checkout counter. In most supermarkets, paper price labels are still manually applied to shelves and display areas. This process leaves much to be desired because it is a costly, labor-intensive, error-prone operation. Thus, electronic price tagging in large-sized markets has gained interest for several years, and a number of systems have been developed around various wireless networking technologies. Yu *et al.* [7] discusses the implementation of the Electronic Intelligent Tag (EIT) system on wireless sensor networks for intelligent management of supermarkets. EIT is an electronic display device that replaces the traditional paper price tag. It also provides a way to distribute frequent and effective promotional activities. Although the paper proposes an architecture and design, it does not provide details of the implementation nor performance evaluations. Furthermore the authors do not discuss wireless communication challenges of the environment as we present in Chapter 3. The work closest to ours is the electronic price label (EPL) system [6], which provides a similar electronic replacement for paper

labels. The EPL system features two-way communication between a controller and electronic price labels to ensure price accuracy and reliability. The authors also measure wireless link performance in terms of the received signal strength and link path loss in a real supermarket environment. Nevertheless, the evaluation of EPL does not consider variability of wireless link characteristics in real-world market environments. They do not provide details on whether they have done experiments in a real supermarket (and during business hours), and they do not discuss the networking aspect nor the delivery performance with respect to time (we run 10 hour experiments during day and night) and burstiness in a crowded human-active environment.

1.2.2 Wireless Systems Community

In one of the earliest work on sensor network architectures, Estrin *et al.* [8] motivate the need for application-specific multi-node aggregation architecture for wireless sensor networks (WSNs). Culler *et al.* [9] describe SNA, a software architecture that describes the principles by which mote software and services are arranged. They also define the “narrow waist” of the architecture to be a translucent sensor protocol layer that exports neighbor management and a message pool on top of which several network protocols can be built [10]. Tenet [11] proposes a tiered architecture for embedded sensor networks which constrains the placement of application functionality in the system. Essentia [12] proposed “asymmetric function placement”, similar in

spirit with Tenet, as a guiding principle to architect sensor network systems. The above works focus on the software architecture of how the functionalities of the sensor network should be organized to overcome the resource constraints of sensor nodes. Furthermore, they do not consider in detail the communication and routing architectures, application requirements specific to market environments, nor real-world measurement of link characteristics in densely populated obstacle-rich large market settings.

Research philosophy of wireless systems community has required researchers to verify their systems' performance through real deployment in the target environment. As results, wireless systems researchers have explored various environments. Juang *et al.* attached low power networking devices to zebras to monitor their behaviors through a wireless network [13]. Mainwaring *et al.* deployed a WSN in Great Duck Island for habitat monitoring [14]. Tolle *et al.* installed a wireless sensing system in redwoods to measure air temperature, humidity, and solar radiation [15]. Werner *et al.* designed and deployed a volcano monitoring system which gathers seismic and acoustic wave information using a WSN [16]. Kim *et al.* designed a wireless system for health monitoring of civil infrastructures and deployed it on Golden Gate Bridge [17]. Ko *et al.* deployed a CTP-based WSN at Johns Hopkins Hospital and showed that multi-hop WSNs can provide reasonable performance for monitoring statuses of patients [1]. However, the market environment still remains unexplored although

it is very close to our everyday lives.

A number of studies have investigated the performance of IPv6 routing protocol for low power and lossy network (RPL) [18] and IEEE 802.15.4 in various network configurations. Ko et al. experimentally evaluated the performance of RPL and 6LoWPAN [19] using TinyOS [20] and showed that the performance is similar to the widely used collection tree protocol (CTP) [21], the *de facto* data collection protocol in TinyOS, while benefiting from an IPv6-based architecture. They also evaluated the performance of ContikiRPL and TinyRPL over uIPv6 and BLIP, respectively [22]. In wireless sensor networks where Contiki and TinyOS are the popular underlying operating systems, they showed that the two embedded IP stack implementations are interoperable but parameter selection and implementation details can have significant effect on the performance of a network consisting of both implementations. Kim *et al.* evaluated the performance of TCP over RPL on a multi-hop low power network testbed [23]. These works revealed that RPL's downwards packet delivery performance is less efficient than the upwards performance. Cisco designed and deployed a field area network (FAN) for smart grids (CG-Mesh) [24], which uses 6LoWPAN, RPL, and IPv6 on top of IEEE 802.15.4 to provide end-to-end two-way communication to each smart metering endpoint. This work verifies that RPL network are feasible in real rural areas and industry will invest RPL-based network solutions. However, it does not consider downward-centric traffic patterns.

Kermajani *et al.* presented simulation results on the network convergence process of RPL over IEEE 802.15.4 multi-hop networks and investigated improvements and trade-offs [25]. Herberg *et al.* compared the RPL protocol with LOAD (6LoWPAN Ad Hoc On-Demand Distance Vector Routing) using NS-2 simulation [26]. They showed that LOAD may incur less overhead than RPL if the traffic pattern is bi-directional. Clausen *et al.* provided a critical evaluation of RPL with respect to limitations and trade-offs, and proposed suggestions for improvements [27]. As simulation studies, the results do not incorporate wireless channel characteristics of the real-world, densely populated market environments.

RPL has drawn significant attention in the smart grid domain and several works have studied the applicability and performance of RPL in this context [28]. Ancillotti *et al.* presented an overview of the role of RPL for smart grid communication and studied ContikiRPL performance using Cooja simulation [29]. Wang *et al.* discussed the use of RPL for AMI in smart grid and compared RPL with AODV routing using NS-2 simulation [30]. Bressan *et al.* discussed the deployment of a smart monitoring system using low power and lossy networks (LLNs) and performed RPL simulations for a smart grid scenario [31]. Although these works provide good overview of how RPL is applicable to the smart grid, they are simulation studies and do not provide evidence of protocol behavior on real devices. Gungor *et al.* measured IEEE 802.15.4 link quality in real power grid envi-

ronments and discussed associated opportunities and challenges [32]. However, their work was limited to the link layer and did not consider the routing or application layer performance.

A number of studies revealed issues related to RPL and tried to alleviate these problems. Ancillotti *et al.* proposed a cross-layering design for RPL, which provides enhanced link estimation and efficient management of neighbor tables [33]. They used AMI as a case study and employed Cooja emulator to evaluate their proposal. The work in [34] 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. Lee *et al.* investigated interoperability problems between RPL and Bluetooth Low Energy (BLE) [35] and alleviated the problems by designing an adaptation layer between RPL and BLE [36]. They used a Broadcom chip for BLE module and implemented the total package of RPL over BLE on Raspberry Pi. Even though these works provided better upward routes, they did not consider downward traffic delivery performance of RPL. In the perspective of downward route management, Ko *et al.* showed that RPL has a serious connectivity problem when two mode of operations (MOPs) are mixed within a single network [37]. To address this issue, the authors proposed DualMOP-RPL that supports nodes with different MOPs to communicate gracefully in a single network

while preserving the high bi-directional data delivery performance. However, this work focused on addressing interoperability problems between two MOPs, rather than finding better downward routes.

1.2.3 Wireless Network Community

Several studies in wireless network community considered asymmetric (heterogeneous) capability among nodes. Ryu and Cho proposed a new routing scheme which makes battery-powered mobile nodes not transmit a packet through a multi-hop route but directly transmit it to the root [38]. Given that the root covers all the nodes in the network via a single hop by using very high transmission power, it directly transmits the received packet to the destination node. DEAR [39] enabled each mobile node to select whether to transmit data packet through the root or a multi-hop route. Liu *et al.* identified that mobile nodes' transmissions suffer from the high-power root's transmission. To solve the problem, they designed DELAR [40, 41] which separates the root's transmissions from mobile nodes' ones in the time domain using a superframe structure. However, these work assumed that all nodes do not have transmission power limit and directly communicate with the root.

Some work proposed topology and transmission power control algorithm which considers link asymmetry. Li and Hou proposed a topology control algorithm under the assumption that each node exploits different transmission power [42]. Lin *et al.* designed each node

to control its transmission power considering wireless network connectivity [43]. Although these works considered link asymmetry among nodes, they still constructed homogeneous networks in the routing perspective. That is, they constructed multi-hop networks in which bi-directional routes have the same hop distance, instead of creating multi-hop upward and single hop downward connectivity.

CLSM [44] and LRPB [45] considered hop distance asymmetry, which means that a low power node is connected to the root node via multi-hop upward and single hop downward routes. They used high power root's transmissions only for broadcasting routing control packets such as RREQ, rather than delivering data packets, to avoid interference from the high power signal. Thus, these work focused on how to eliminate data packet transmissions from the root node. In contrast, we actively use high transmission power of the root node for reliable data packet delivery.

Lastly, some studies such as MC [46] and TACP [47, 48] considered using high power root's transmission for data delivery. They include the root's high power capability in multi-hop route discovery, which significantly reduces hop distance. However, since a low power node which is distantly located from the root cannot send an acknowledgment (ACK) packet to the root, it transmits the ACK in a multi-hop manner. This results in significant transmission overhead.

1.3 Contributions and Outline

In this dissertation, we incorporate real-world networking constraints by taking a practical approach and design a prototype system for wireless e-price tagging applications. Specifically, our design and implementation addresses (1) delivery of downstream-focused traffic through wireless network for e-price tag updates, (2) automatic repeat request (ARQ)-based reliability considerations with minimal operational overhead, (3) low and fair energy consumption at price tag nodes, and (4) IP compliant network architecture for interoperability and usability.

We first investigate wireless link characteristics in an urban, crowded large-scale indoor market place of dimension 90×60 meters which displays more than 10000 items and is visited by over 5000 customers per day (Figure 1.2). From this preliminary pilot study we identify that large crowded markets have external noise sources, active human movements, and some market-specific activities that heavily impact the link-level performance of wireless systems. Furthermore, we perform preliminary deployment of a prototype system, which comprises the IETF standard IPv6 routing protocol for low-power and lossy networks (RPL) [18] and low power listening (LPL) protocol, using low-power embedded networking platforms. We evaluate its performance and reveal that pre-existing network protocols cannot overcome harsh wireless channel in crowded markets.



Figure 1.2: Our experimental market environment: a crowded indoor market with a size of 90×60 meters, $>10k$ types of items, and >5000 customers per day.

With these findings, we design *MarketNet*, a system to address such unique challenges of a busy market. Our design follows an asymmetric transmission power network architecture in which a basestation node can reach individual e-price tags via single hop (using high-power transmission), while the e-price tags transmit data to the basestation over multiple hop links (using low-power radios). We then evaluate *MarketNet* in two different real-world environments. First, we construct a 30-node indoor testbed to validate our proposed system architecture. Following initial validation, we move to the target market

environment to confirm that real-world channel conditions are effectively mitigated by *MarketNet*. We also use RPL-based network for benchmark comparison. From these deployments and experimental evaluations, our results show that *MarketNet* adapts well to real-world wireless channels, significantly improves reliability, and maintains a radio duty-cycle of about half of RPL-based network in most cases.

Specifically, the contributions of this dissertation are four-fold.

- First, we introduce application-level requirements and technical challenges in designing an e-price-tag system for indoor markets, collected from a series of interviews with store managers.
- Second, we investigate wireless link characteristics and empirically measure the performance of current *de-facto* IPv6 standard network protocol in a real-world crowded indoor market place. From this preliminary study, we identify unique wireless challenges and performance issues in market environments.
- Third, with the application requirements and real-world challenges at the basis, we design *MarketNet* for wireless e-price tagging in large crowded market environments. Our *MarketNet* includes a set of key ideas as follows: (1) an asymmetric transmission power-based network (APN) which comprises single hop downlink transmission by using a high transmission power root and multi-hop uplink transmission by using RPL and low power nodes, (2) local retransmission by neighbor nodes, (3)

a network-wide superframe architecture, and (4) uplink period partitioning.

- Lastly, we validate the performance of *MarketNet* under various environments including a large, real-world indoor market with 10000+ items and 5000+ customer base. Our results show that APN-based tagging performs well under real-world channel conditions.

This dissertation is structured as follows. In Chapter 2 we introduce the concept of price representation, our application scenario for e-price tagging, and its requirements. Next, Chapter 3 presents results from our preliminary study to better understand wireless channel environments of crowded markets and identify practical design issues. We also perform our initial pilot deployment of a wireless price tagging system in an indoor market using an existing network protocol. Chapter 4 introduces our initial design for e-price tag system, *MarketNet1.0*, and extensively evaluates performance using mathematical analysis, computer simulation, testbed experiments, and real market deployment. From the experience of *MarketNet1.0*, we identify additional challenges that we need to overcome to fulfill application requirements. Chapter 5 introduces an advanced design *MarketNet2.0* which alleviates problems of *MarketNet1.0*. We also extensively evaluate its performance using testbed experiments and real market deployment, which shows that *MarketNet2.0* successfully addresses envi-

ronmental challenges and provides acceptable performance to support e-price tagging application. Lastly, we conclude the work with a summary and further research direction in Chapter 6.

Chapter 2

Target Application: Wireless and Remote Update of e-Price Tags

This chapter motivates the electronic price tagging application and describes its requirements.

2.1 Price Representation

For shoppers, price representations are simply responses to their questions, ‘How much is it?’ [49]. Defined more formally, price representations are the various ways in which prices are made available to market participants or renderings of prices generated and disseminated by market actors [50]. Sellers produce and post price representations

in order to make prices available to potential buyers whereas buyers and third parties offer them in order to make price comparisons or track price changes. Particularly in the retail context, the practice of producing price representations can be complex and ambiguous due to a few reasons [50]. First, prices change over time because of their strong signaling effects. Second, retailers strategically use different prices of the same good to affect their sales. Third, the method to represent prices determines the representing efforts.

While innovations in retail services span a broad spectrum of initiatives [51], innovations in price promotions that involve both process and technology can create values (specifically, increase revenue and profit) by providing considerable purchase opportunities to target customers or shoppers effectively through both offline and online [52]. Recently, some marketers have employed dynamic pricing models that update prices frequently by using data from online and offline purchases or company enterprise resource planning systems to set prices based on changing supply or demand characteristics [53]. For example, retailers drop prices when the user base is below a target level, and increase prices when the user base is above the target [54]. They can also adjust their prices based on the prices of their competitors. Particularly, time-based pricing allows retailers to adjust prices according to how long a product has been on the market or the time of day. They may increase the demand for an older product by marking it down [54, 55].

Although dynamic pricing can improve profit in an ideal case, when applied to real-world markets, it causes some side effects which may decrease profit. First, represented prices can be different from real ones due to human errors, which incurs considerable customer complaints and profit loss. Second, frequent reconfiguration of price tags in large scale markets requires significant labor cost. To confirm this burden, we conducted interviews with market managers and they indicated that price updates happen frequently. Prices are updated with respect to the products' freshness (e.g., meat, bread, produce) or due to real-time pricing of competing markets. Furthermore, they also indicated that, currently, price updating occurs manually and customer complaints due to incorrect prices are one of the major challenges in large-scale markets. Given such practical operational burden in large-scale marketplaces, an electronic system that automates the price updating process can benefit the market and improve the quality of service for the customers.

2.2 Application Scenario

Our target application, a wireless system for remotely updating e-price tags, aims to automate the price tag reconfiguration procedure for large and crowded market environment as pictured in Figure 1.2 by using low power and low cost wireless embedded system technologies. Specifically, this electronic price-tagging system targets to max-

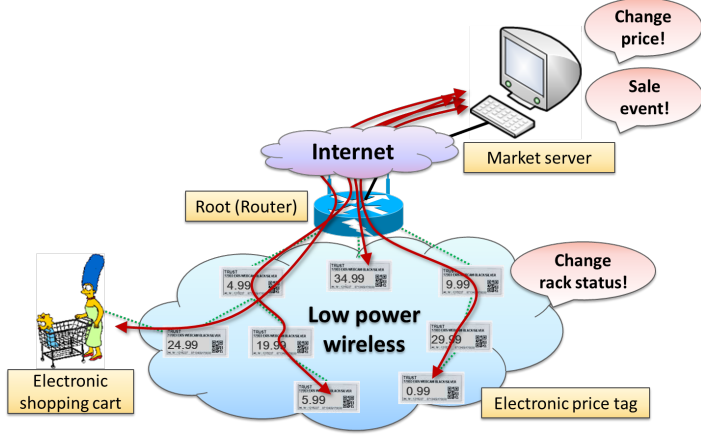


Figure 2.1: Application scenario for our wirelessly reconfigurable e-price tagging system, consisting of a wireless-enabled central server and end devices such as price tags and shopping carts.

imize profit by reducing labor cost and customer complaints as well as motivating customers to purchase more products. Consisting of a low power embedded computing unit accompanied with a low power wireless radio and a low power display module (e.g., electronic-ink), e-price tags are capable of displaying the prices for a specific product, while receiving real-time updates on price changes from a central server computer.

On a comprehensive system-level perspective, an e-price tag system will consist of many electronic tags deployed at the shelves where the items are stored. Each e-price tag is connected with the central server through wireless networks, which allows them to remotely exchange information each other.

We illustrate the functionalities of e-price tag system using a dia-

gram in Figure 2.1. Upon changes in the prices, with respect to the market management policies, the central server, which is controlled by the retailer, will send updated price information to assure that the displayed prices will always present up-to-date information. Given frequent price updates in large-scale retail stores, this automation can greatly reduce labor cost which has been paid for redundant manual price updates. Furthermore, on a management perspective, this change reduces the chances of man-made errors and delayed updates, which naturally leads to a reduction in profit loss at markets due to customer complaints.

In addition to price updates, we can envision that these tags may also have the capability to track the quantity of items remaining on the shelves (e.g., using weight or infrared ray sensors). Using the wireless radios from which the tags receive updates, such additional information can be periodically sent to the central computer so that market managers can keep track of stock in real-time. With this inventory monitoring service, market staffs do not have to go around the whole markets to check the status of each single item, which results in labor cost reduction. Moreover, they can restock each product before it runs out, which prevents customers giving up purchase of products due to lack of inventory and provides additional profit.

The same wireless infrastructure, once available, can easily extend to other intra-market applications. As an example, with wirelessly controllable display units on shopping carts, the retailer may provide

advertisements for the shoppers when they show interest in products that are located in a specific area or when discounts and promotion occur, which motivates them to move and purchase. Wireless infrastructure can deliver the advertising information farther than voice announcement.

2.3 System Requirements

For this target application, our work began with a series of interviews with market managers where we gathered a set of application level requirements as below.

- **Downstream-focused Traffic:** For e-price tagging, a majority of the traffic will be price updates that occur several times a day, along with promotional information updates for shopping carts from a central server. Most of the upstream traffic from tags to the server are acknowledgments and rack status updates. Overall, bottlenecks are more likely to arise in the downstream direction. Since most wireless sensing systems, which have been designed and deployed, target to support monitoring applications which generate upstream-focused traffic, this requirement is quite new and not trivial.
- **Mass-scale and Real-time Price / Status Updates:** For updating price tags, multiple products may require reconfiguration, or the promotion information on many carts may require simultaneous updates. Therefore, the system should effectively support

mass-scale updates, specifically at least 100 products per minute, as indicated by market managers (and also in previous work [6]). Furthermore, managers requested that the system makes price updates within 10 seconds after updating the server.

- **Data Reliability:** Market managers selected reliability of product price updates as the highest priority system requirement. A system for e-price tagging should assure that the prices are reliably updated with minimal transmission overhead by successfully overcoming environmental challenges. Reliability of the wireless system should be at least higher than that of manual updates and able to address the ill-defined problem of price presentations. Since a marketplace is a human-dense environment in many situations, this naturally complicates the wireless environment. Resolving such issues not only requires careful system deployments (e.g., locations of where each device is installed), but also improvements at the functionality level, in other words, wireless protocols.
- **Low Labor Costs:** A price-reconfiguring system in a market should be robust and also be easily deployable by market staff. For this, a full wireless system (without the need for power cable extensions) is needed. This suggests that the devices in the markets should make use of battery-based power for their operations. Furthermore, given that battery replacement is another source of labor cost, the devices are required to consume low energy to prolong

battery lifetime.

- **Wireless Nodes with Low Cost and Low Energy Consumption:**

While wireless communications such as LTE modems are widely used, they suffer from high energy usage and cost issues. A more affordable and sustainable approach would be using radios in the industrial scientific medical (ISM) band, such as WiFi. But, again, even for WiFi chipsets, achieving low energy is a significant challenge. The wireless module used in our application should be cheap, maintain a long battery lifetime, and minimize the deployment and management costs. While the root node’s energy consumption is less of an issue, for the system to be practically useful, the lifetime of the e-price tags’ and shopping carts’ radio modules should last for >3 months¹. For energy efficiency, an e-ink-based display is preferred over LCD, and radio duty-cycling is a must.

- **Usability:** Our system should be designed so that it is easy for the market staffs to utilize and manage. Given that markets typically have network managers who are familiar with IP protocols, underlying wireless network of our system needs to be IP compliant.

¹While a 3 month deployment may seem short for a low power embedded node (considering systems deployed in outdoor environments), our surveys show that this is within tolerable range for the market managing staff. Achieving a longer lifetime would be beneficial, but, we surveyed the staff for a minimum lifetime that they could tolerate given the hardware constraints of our system. Note that, the sales items on the shelves need to be re-stocked manually by employees several times a day. Therefore, replacing a pair of AA batteries every 3 months adds only minimal extra labor. Thus, this is still a several orders-of-magnitude improvement to current practice with paper tags.

Chapter 3

Preliminary Study in Urban Crowded Markets

3.1 Introduction

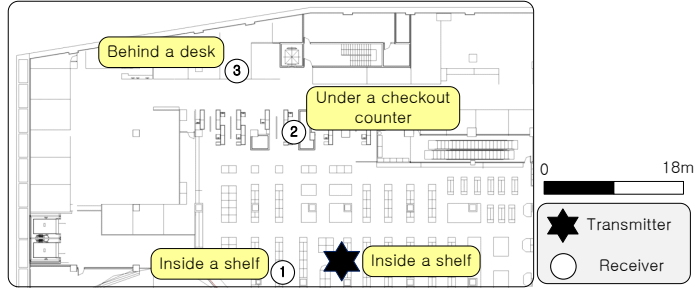
As a first step to design a wireless system for market environment, while satisfying the application requirements, we performed a preliminary study to understand the wireless environment in real market environments. Research philosophy of wireless systems community has required researchers, who want to design a system in an unexplored environment, to verify their systems' performance through real deployment in the target environment. As results, wireless systems researchers have explored various environments such as island, wildlife, volcano, woods, bird nest, bridges, hospitals, and rural areas as in Figure 3.1 [1, 13, 14, 15, 16, 17, 24]. However, the market environment



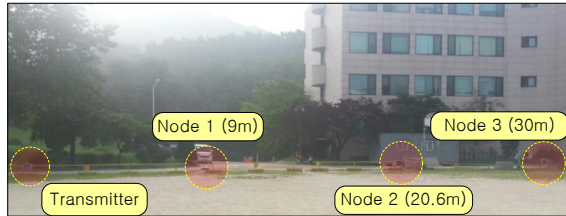
Figure 3.1: Various environments that have been explored by wireless systems researchers. From the upperleft, island, wildlife, volcano, bird nest, woods, bridges, hospitals, and rural areas.

still remains unexplored although it is very close to our everyday lives.

In this chapter, we experimentally investigate wireless link characteristics in an urban market environment. This measurement study reveals that urban crowded markets incur considerable path loss due to obstacles such as walls, products, and metal shelves. Furthermore, they have significant link dynamics due to not only general human activities such as movements and WiFi usage, but also some market-specific activities such as item refilling events and microwave oven usage. Next, we deploy a representative multi-hop network protocol in the marketplace and investigate whether it can fulfill our application requirements or not. Our performance evaluation shows that existing multi-hop network protocols are not suitable to support our e-price tag application. Overall, the preliminary pilot study in this chapter motivates us to design a new wireless system for remotely



(a) Market topology



(b) LOS topology (playground)

Figure 3.2: Link testing environment.

reconfiguring e-price tags.

The rest of this chapter is structured as follows. In Section 3.2 we investigate link characteristics in an urban crowded market environment. We evaluate performance of a pre-existing multi-hop network protocol in Section 3.3 and summarize our results in Section 3.4.

3.2 Wireless Channel Characteristics

To understand the wireless conditions in our target environment, we first performed a study to investigate channel environments in various dimensions. For this, we set up a testing environment as in Figure 3.2(a). We install three receivers with a single transmitter



Figure 3.3: Low power node for measuring wireless link characteristics in markets.

broadcasting packets of 72 bytes (MAC payload) at an inter-packet interval (IPI) of 50 msec with 0 dBm transmission power. As depicted in Figure 3.3, each node is a TelosB-clone device [56] which combines MSP430 microcontroller [57] with TI CC2420 transceiver [58] and has an antenna gain of 5 dB. Furthermore, it has a Raspberry Pi to record real-time log messages and an external battery for long term measurement. We select the receivers' locations so that we capture various aspects of the market including active human movements, RF propagation over metal shelves and long-distance communications. Unless specified, we take our measurements on IEEE 802.15.4 channel 26, which, in the U.S., is free from WiFi. For reference, we conducted another line-of-sight (LOS) experiment, where each link has the same distance as the market, but in a different environment (i.e., playground in a campus) as in Figure 3.2(b).

Figure 3.4 presents various link-performance metrics from this ex-

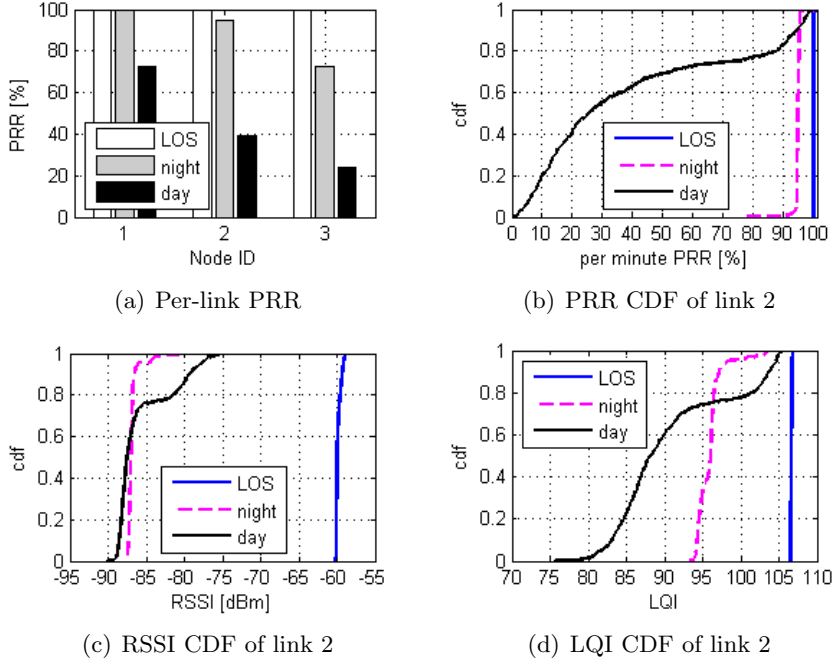


Figure 3.4: Link characteristics of our indoor market environment for channel 26. Packet delivery performance degrades due to the effect of metal shelves and human activities during the day-time.

periment. First, Figure 3.4(a) presents the per-link packet reception ratio (PRR) both for the day-time (noon-10PM) and night-time (11PM-9AM). The performance of wireless links measured in the market differs significantly when compared with the LOS cases, which validates that various factors in a market environment can indeed complicate the wireless environment. Specifically, the fact that PRR of market night-time cases is lower than that of LOS cases reveals that walls, products, and metal shelves carrying items in the market make it an obstacle-rich environment and degrade wireless perfor-

mance even without human activities. Moreover, by comparing the day- and night-time cases within the market, we can see that the majority of the performance degradation comes from human activities throughout the day-time.

To understand the impact of human activities further, we analyze the characteristics of link 2 in detail, where the receiver is located at one of the most crowded areas in the market. Figures 3.4(b), 3.4(c) and 3.4(d) plot the cumulative density function (CDF) of the per-minute PRR, received signal strength indicator (RSSI) and link quality indicator (LQI) of link 2 for LOS, day- and night-times. These figures show that all performance metrics in the market are significantly worse than the LOS case: suggesting that an urban market introduces a challenging wireless environment. Furthermore, combining Figures 3.4(a) and 3.4(b) shows that, not only is the average PRR for the day-time simply lower than the night-time, the per-minute PRR for the day-time is widely spread from 0% to 100%. The fact that per-minute PRR during the night is mostly $>95\%$ serves as evidence that day-time human activities not only degrades the average packet delivery performance, but also gives heavy impact on the link's dynamics. Other metrics such as RSSI and LQI show a similar trend, in which the values are much more dynamic during the day-time.

It is meaningful to point out that the average RSSI shown in Figure 3.4(c) is higher during the day- than the night-time despite its low PRR. We explain this using Figure 3.5, where we plot the channel

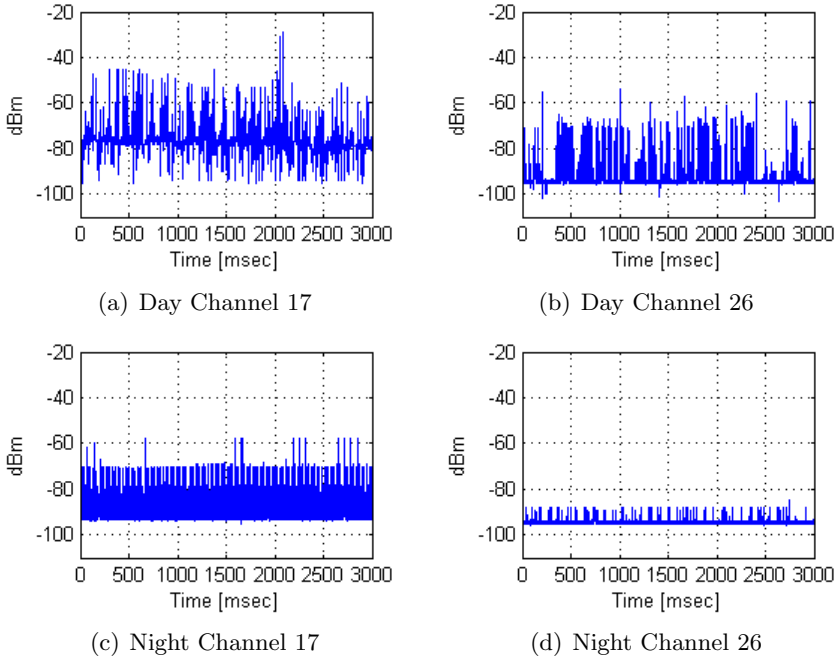


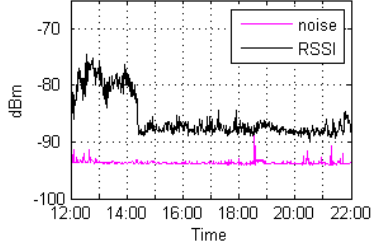
Figure 3.5: 1 kHz RSSI sampling traces. The country we test in (South Korea) allows for WiFi traffic on channel 26, causing noise spikes during the day.

noise in the market sampled at 1 kHz. Notice here that there is a substantial amount of noise on channel 26, comparable enough to channel 17, which actively interferes with WiFi traffic. After some investigation, we identified that there was a significant amount of WiFi traffic on channel 26 as well, which is authorized in South Korea, where our experiments were conducted. Therefore, active WiFi on channel 26 caused the day-time background noise levels to increase. While many sensor networking protocols utilize RSSI as an easy-to-gather, low-complexity, and robust networking metric [59], our results imply that

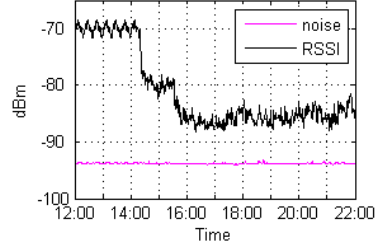
using RSSI in our target environment is not a good design choice in selecting high quality links.

Figures 3.6(a) through 3.6(d) plot the per-minute average RSSI and noise floor levels of links 1 and 2 for 10 hours during the day- and night-times, respectively. From these figures, we can see similar phenomena as well. The long term measurement plots provide us with a longitude perspective of the short-term noise sampling presented in Figure 3.5. Moreover, we can observe that per-minute average RSSI continuously fluctuates during the day-time, which may come from general human activities such as movements and WiFi usage.

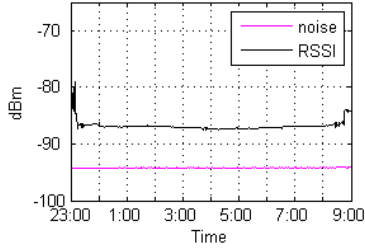
Furthermore, we make another interesting observation which cannot be caused by such general human activities. Both links 1 and 2 experience a sudden and dramatic decrease in RSSI at the same time (between 2PM and 3PM) and link 1 experiences another sudden drop between 3PM and 4PM. Once the RSSI level dropped, this lower RSSI was continuously maintained throughout the day. To get the reason of this phenomenon, we kept watching the shelves where we deployed the nodes, and observed that a market staff refilled the item on the shelf where the transmitter is deployed, between 2PM and 3PM, as in Figure 3.7. Thus, we can confirm that the refilling event (i.e., water bottles) on the shelves at which the transmitter was located incurs sudden RSSI drop for both links 1 and 2 simultaneously. From this observation, we can infer that the reason why only link 1 experiences another sudden RSSI drop is that another refilling



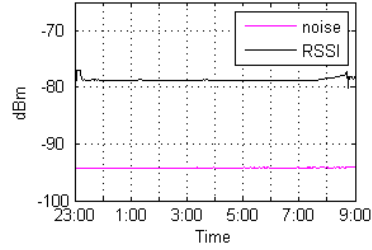
(a) Daytime RSSI/Noise of link 2



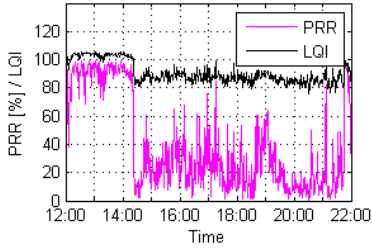
(b) Daytime RSSI/Noise of link 1



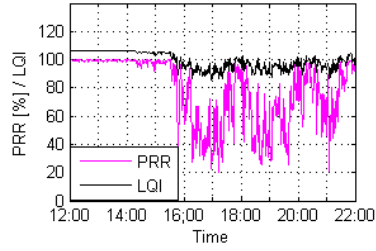
(c) Nighttime RSSI/Noise of link 2



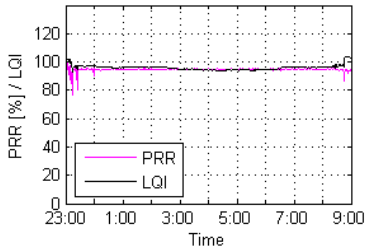
(d) Nighttime RSSI/Noise of link 1



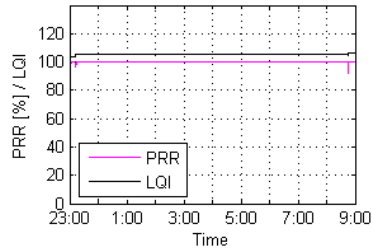
(e) Daytime PRR/LQI of link 2



(f) Daytime PRR/LQI of link 1



(g) Nighttime PRR/LQI of link 2



(h) Nighttime PRR/LQI of link 1

Figure 3.6: Characteristics of links 1 and 2 over time for channel 26. General human activities and some market-specific activities cause links to fluctuate on both short and long term perspectives.



Figure 3.7: An event of refilling items on a shelf, performed by a market staff. Some market-specific activities can change wireless environment and cause long term fluctuation on link quality.

event occurred on the shelf at which the receiver node 1 was located, between 3PM and 4PM. We can conclude that our target environment introduces many man-made hard-to-predict challenges (similar to those reported in [60, 61]), which are unique to the indoor market environment, causing the wireless links to fluctuate on both short- and long-term perspectives.

Figures 3.6(e) through 3.6(h) plot the per-minute PRR and per-minute average LQI of links 1 and 2 for 10 hours during the day- and night-times, respectively. These figures show that this same artifact impacted the LQI and PRR performances as well. Additionally, we can observe that PRR of link 2 degraded after link 2 experiences the

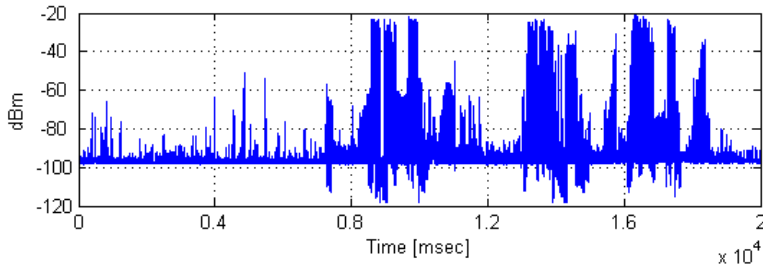


Figure 3.8: 1 kHz RSSI sampling on channel 26, performed near a microwave oven. Microwave oven is another major factor of link quality fluctuation.

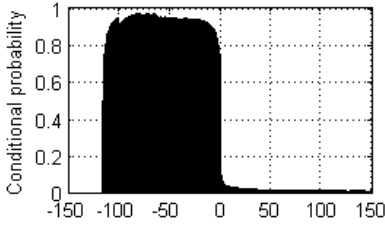
RSSI drop (between 2PM and 3PM), but PRR of link 1 degraded after link 1 experiences two RSSI drops (between 3PM and 4PM). Given that link 1 has shorter distance than link 2, we can confirm that the combination of different distances, general human activities, and market-specific activities further complicates wireless environment.

Another market-specific activity is handing out food samples. Microwave ovens, which are well known wireless interferers in 2.4 GHz ISM band [62], are frequently used to provide samples of cooked food at several points in the targeted market. To confirm the impact of microwave ovens, we use Figure 3.8, which plots the 1 kHz noise sampling traces for when a near-by microwave oven is active in the market. We noticed that the microwave oven also heavily impacts the wireless link characteristics.

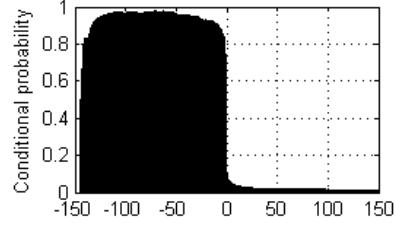
Lastly, we examine the packet loss patterns of link 2 by analyzing the conditional packet delivery function (CPDF) in Figure 3.9. The CPDF (introduced in [63]) corresponds to the probability of a

packet being successfully received after n consecutive failures or successes. Negative numbers represent consecutive successes, while positive numbers represent consecutive failures. For example, $CPDF(20)$ is the probability of a successful delivery after 20 consecutive failures on the link. Likewise, $CPDF(-20)$ is the probability of a successful reception after 20 consecutive successful receptions. Therefore, the CPDF is a good measure of link burstiness and the channel coherence time.

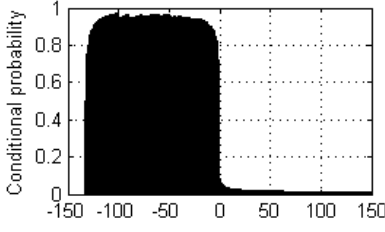
Figures 3.9(a) through 3.9(g) sequentially plot the per-hour CPDF of link 2 from 3PM to 10PM, after the refilling event occurred at the rack near the transmitter between 2PM and 3PM. We can first notice from the non-uniformity of the CPDF plots that links in the market environment are heavily bursty. Furthermore, since the length of the CPDF's negative (e.g, left) tail represents the maximum number of consecutive successes (or maximum time duration of good link quality, when combined with IPI), we can see here that the link burstiness varies over time. We conjecture that this was an impact of human movement activities, and to validate this we present the correlation between the negative tail length (i.e., maximum consecutive successes per hour) and human movements (manually collected on the link between the transmitter and receiver node 2) in Figure 3.9(h). Here we validate the fact that the positive burstiness decreases with an increasing number of customers, naturally suggesting that human activities impact the link burstiness.



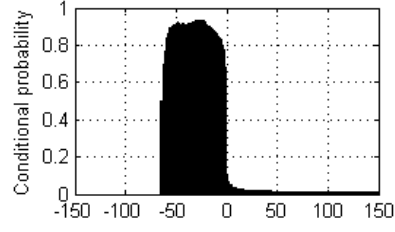
(a) 3~4PM



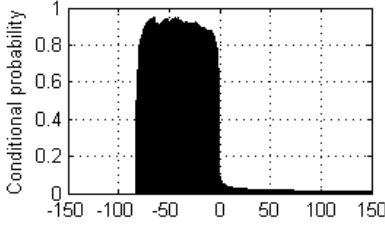
(b) 4~5PM



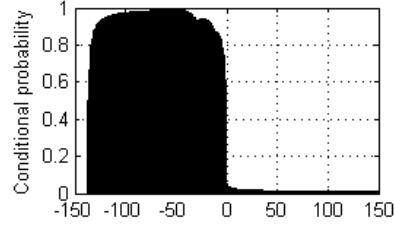
(c) 5~6PM



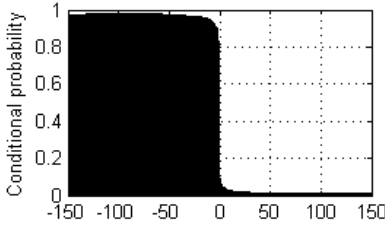
(d) 6~7PM



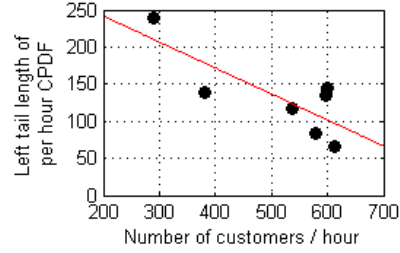
(e) 7~8PM



(f) 8~9PM



(g) 9~10PM



(h) Correlation between human population and CPDF

Figure 3.9: Per hour conditional packet delivery function (CPDF) for different hours during the day, along with the negative tail (or left tail) lengths' correlation with human population. The negative tail length (i.e., positive burstiness) decreases with the number of customers.

3.3 Performance of Pre-existing Protocols

From the results in Section 3.2, our next question is, “*Can pre-existing multi-hop network protocols overcome such link dynamics in the market environment?*” To answer this question, we deployed a prototype system using an existing network protocol in an urban crowded market and evaluated its performance. Specifically, we exploit RPL [18] and low power listening (LPL) protocols to fulfill the two of application requirements; low energy consumption and IP compliant network. RPL is IPv6 standard routing protocol for low power and lossy network (LLN) and LPL is the default link layer protocol in TinyOS. We first describe the operations of RPL and LPL, and then, present the results from performance evaluation of RPL and LPL in an urban crowded marketplace.

3.3.1 RPL Operation

In this subsection we describe TinyRPL, i.e., the default RPL implementation in TinyOS 2.1.2 (latest), which implements the RPL standard [18] with OF0 along with the hop count metric for rank calculation and the ETX for parent selection.

RPL broadcasts the routing information using DODAG information object (DIO) messages which are transmitted based on the *TrickleTimer* [64] to achieve a balance between control overhead and fast recovery. To this end, the *TrickleTimer* doubles the broadcast period

after every DIO transmission and re-initializes it to a minimum value when route inconsistency is detected. Furthermore, *RANK* is defined and used by the OF to represent the routing distance from a node to the LBR, and link and node metrics are used for *RANK* calculation and parent selection.

TinyRPL with OF0 uses *hop count* for *RANK* calculation, and together with *ETX* for parent selection. Specifically, *RANK* of node k is defined as

$$RANK(k) = h(k) + 1 \quad (3.1)$$

where $h(k)$ is the hop count between node k and the LBR. That is, $RANK(LBR) = 1$, and $RANK(k) = \infty$ before node k joins the network. Node k broadcasts DIO messages containing $RANK(k)$. $ETX(k, p_k)$ measured by node k is a link quality indicator between node k and its parent candidate p_k , and is defined as

$$ETX(k, p_k) = \frac{\# \text{ of total transmissions from } k \text{ to } p_k}{\# \text{ of successful transmissions from } k \text{ to } p_k}. \quad (3.2)$$

RPL smoothes the *ETX* using an exponentially weighted moving average (EWMA) filter, making it robust to sudden changes in link condition.

Each node recognizes its neighbor nodes by DIO messages received from them. Node k generates its parent candidate set \mathbf{P}_k from its

neighbor set \mathbf{N}_k as

$$\mathbf{P}_k = \{n_k \in \mathbf{N}_k \mid h(n_k) < h(k), \text{ } ETX(k, n_k) < \delta\} \quad (3.3)$$

where δ is a threshold to remove neighbors which are connected through unreliable links.

Each node performs parent selection process when its information on parent candidates has been changed. Node k selects its best alternative parent \hat{P}_k as

$$\hat{P}_k = \arg \min_{p_k \in \mathbf{P}_k} \{R(p_k)\} \quad (3.4)$$

where $R(p_k)$ is a routing metric given as

$$R(p_k) = RANK(p_k) + ETX(k, p_k). \quad (3.5)$$

Then, it changes its parent node from the current parent P_k to the best alternative \hat{P}_k if

$$R(\hat{P}_k) < R(P_k) - \sigma \quad (3.6)$$

where σ is a stability bound to mitigate unnecessary and inefficient parent changes, which is set to 0.5 by default. This is a hysteresis component (similar to MRHOF) of TinyRPL, and we refer to it as the stability condition.

RPL constructs downward routes simply as the reverse of upward ones. In RPL, each node sends a destination advertisement object (DAO) message towards the root periodically¹ and also when its upstream route has changed. TinyRPL implements the storing mode of RPL, and thus each node sets up a downstream route to the DAO sender and adds it to the routing table whenever receiving a DAO message.

We rectify a problem is in downward routing of TinyRPL. In RPL, each entry in the downstream routing table is removed when no DAO is received from the destination of the entry for a certain timeout period (20 minutes by default). To this end, TinyRPL uses a timeout counter called *RemoveTimer* for each downstream route entry. However, it does not reinitialize *RemoveTimer* of each entry even when a corresponding DAO message is received. Thus, a new downstream route entry has a fixed lifetime, and a node suffers from the absence of a downstream route between the timeout removal and the reception of next DAO message. We were alerted to this problem in experiments by observing very poor downstream delivery performance. We fixed the problem by re-initializing *RemoveTimer* at every reception of an updated DAO message. The experiments and measurement results reported below were obtained after correcting the problem in order to focus our work on the high-level characteristics of RPL rather than its current implementation.

¹Depending on the implementation, it can be pseudo-periodic. The RPL standard RFC6550 does not mandate the transmission timing of DAO messages.

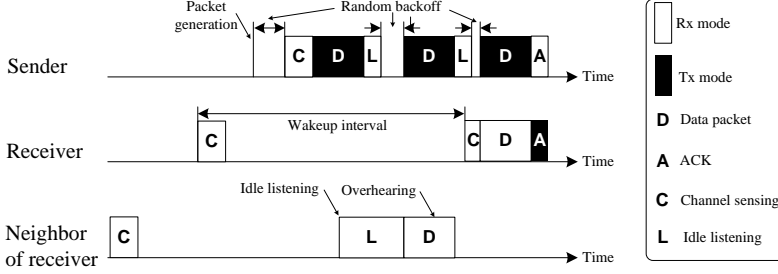


Figure 3.10: An example of BoX-MAC-2 operation. BoX-MAC-2 achieves asynchronous packet delivery by exploiting periodic wakeup at the receiver and repetitive packet transmission at the sender, incurring a trade-off between a sender and a receiver in energy consumption according to the sleep interval.

3.3.2 LPL Operation

This subsection describe the LPL operation based on BoX-MAC-2 [65], which is the default link layer protocol of TinyOS and is widely used in various LLN systems [66, 67]. Figure 3.10 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, 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 sleep 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., $(t_{ack} + t_{bo,max})$ 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 3.10, the receiver node receives and acknowledges the packet in the second sleep period. As we can see in the “neighbor of receiver” case in Figure 3.10, 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 data packet.

This repetitive transmission of a sender and the periodic wakeup of a receiver lead to a successful packet delivery without synchronization. If the sender fails to receive an ACK, it retries to transmit the packet up to a maximum number of n_{tx} times²

BoX-MAC-2 has a trade-off relation between a sender and a receiver in energy consumption based on the sleep interval. As the sleep interval increases, the sender consumes more energy due to the

²One packet retransmission comprises multiple repetitive transmissions during a single sleep interval.

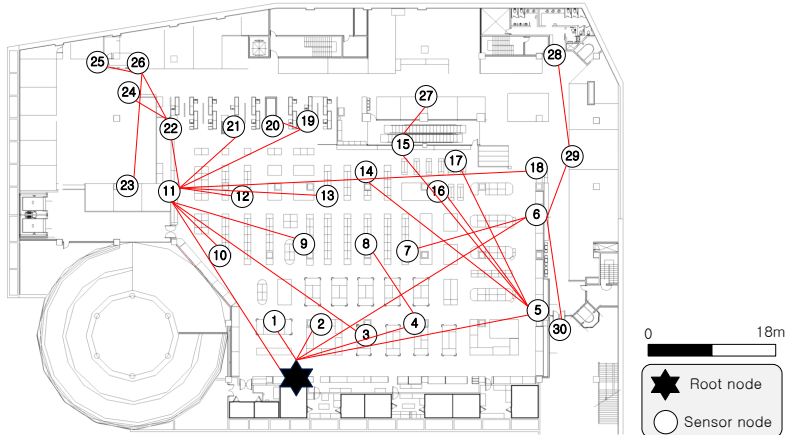


Figure 3.11: Topology map of our 30 node market environment with a snapshot of RPL routes.

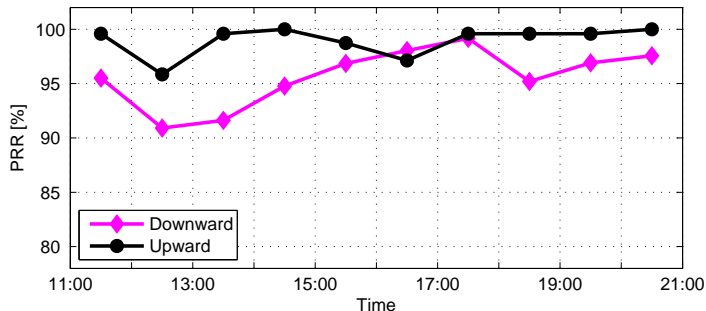
increase in the number of repetitive packet transmissions, while the receiver consumes less energy owing to the reduced frequency in channel sensing.

3.3.3 Performance of RPL over LPL

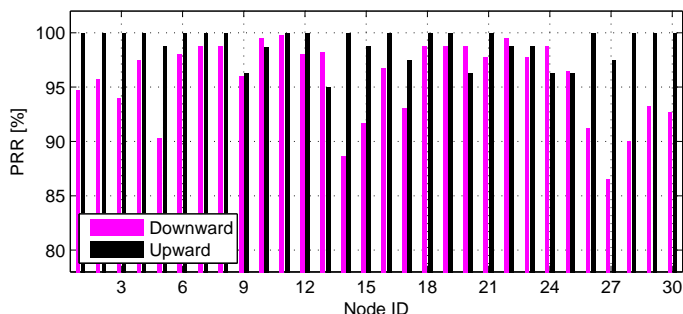
To evaluate the performance of RPL over LPL, we configured a testing environment in the market as in Figure 3.11. We deployed 30 low-power nodes representing reprogrammable price tags, and one base-station (root) node at a single floor market with an area of $90\text{m} \times 60\text{m}$. We point out that the goal of this performance evaluation is to confirm whether the representative low-power multi-hop network protocol (RPL on top of LPL) performs well in real environments, in terms of reliable downstream traffic delivery and energy consumption, when supporting downward-focused traffic. To this end, we consider

a downward traffic-focused scenario where the interval of packet generation (IP packets with 20 byte payload) to and from each node are 90 and 450 seconds, respectively. Given our topology, from the basestation’s perspective, this corresponds to sending one downwards packet every three seconds, and receiving one upwards packet every 15 seconds. For low-power nodes, we use the same device as in the link measurement study with transmission power of -15 dBm, which allows RPL to produce a 4-hop network in our topology. We set the sleep interval of underlying LPL as 2 seconds. We performed the experiments for 10 hours during the day-time (from 11AM to 9PM).

Figures 3.12(a) and 3.12(b) plot PRR performance of RPL over LPL both for uplink and downlink. Figure 3.12(a) shows that while the uplink performance of RPL is satisfying, the downlink performance is not reliable and almost always lower than uplink performance. Furthermore, downlink performance fluctuates more over time than uplink performance, which shows that RPL’s downlink is weak for link dynamics in market environments. Figure 3.12(b) shows that, for almost all nodes, downlink PRR is worse than uplink PRR. Additionally, RPL’s downlink suffers from severe PRR unfairness among nodes. Given that this low and unfair downlink PRR is the result of 10 maximum retransmission efforts at link layer, we can confirm that RPL cannot provide reliable downlink performance when delivering downward-centric traffic in urban market environments. For our application scenarios with electronic and wireless price updates, this can



(a) PRR vs. Time

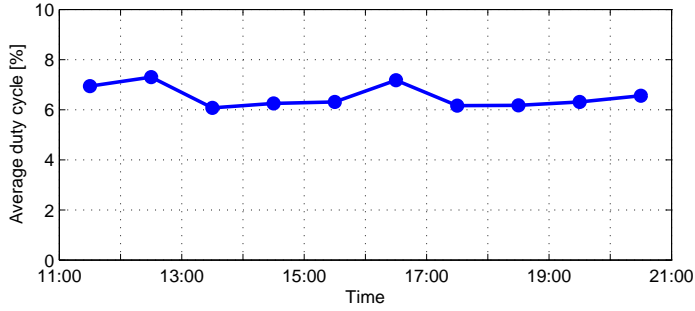


(b) PRR vs. Node ID

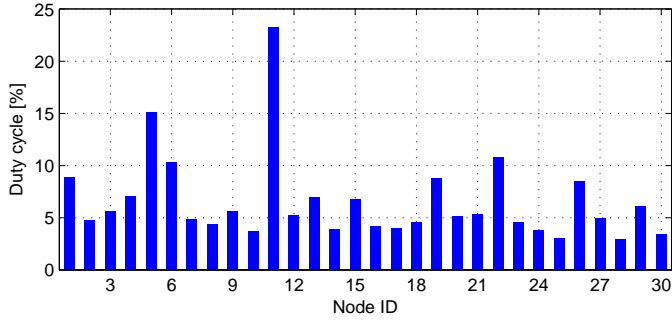
Figure 3.12: Packet delivery performance of RPL over LPL, both for uplink and downlink.

be a critical issue that must be addressed.

Ironically, RPL *was* designed to support an LLN which *has* link dynamics, which is contradicted by our results. After deeper investigations on the reasons for the downwards traffics high loss rate, we were able to identify that RPL was designed to better optimize to deliver upward-centric traffic patterns. As aforementioned in Section 3.3.1, in RPL, only children nodes can initiate route changes. RPL allows them to estimate link quality based on upward traffic delivery and op-



(a) Duty cycle vs. Time



(b) Duty cycle vs. Node ID

Figure 3.13: Duty cycle performance of RPL over LPL.

timize their upward routes using this link quality information. Thus, when delivering downward-focused traffic (i.e., sparse upward traffic), RPL updates link quality slowly and loses many packets due to slow route recovery in dynamic market environments.

Figures 3.13(a) and 3.13(b) plot duty cycle performance of RPL over LPL both for uplink and downlink. Figure 3.13(a) shows that RPL over LPL experiences some level of duty cycle fluctuation over time. More importantly, Figure 3.13(b) reveals that its duty cycle performance is severely unfair among nodes. Specifically, the worst

node (node 11) consumes about 8 times more energy than the best node (node 28). Given that energy fairness is a quite important performance metric since human intervention interval (i.e., labor cost) is directly related to when the first dead node occurs, this unfairness can be a significant problem when supporting our target application.

This unfair energy consumption could be inevitable in multi-hop networks since nodes near the root have to relay more packets. However, we observed from Figure 3.11 that this unfairness is not natural but problematic. It shows that node 11 has 15 subtree nodes which are half of the all nodes, which forces it to suffer excessive relay burden and consume much larger energy than other nodes. In fact, RPL has load balancing problem [34] since it allows each child node to select its parent node considering hop distance and link quality rather than traffic load. Thus, we may improve energy fairness by better protocol design.

3.4 Summary

Based on the observations in the previous section, we summarize unique technical challenges of the market environment as below:

- **Human Movements and Market Activity:** Human activity is one of the main reasons for link dynamics during the day. Specifically, the impact of human movements prevents the links from making successful transmissions for long durations. Furthermore,

the impact of typical market activity, such as the use of the microwave oven, causes market’s link conditions to continuously fluctuate. These issues create the need for a robust networking layer.

- **Metal racks and items:** Metal racks and their items impact the link characteristics in two ways. First, they become an obstacle for RF signals. Second, the item stock status cause long term fluctuation on the link quality.
- **Noise Statistics:** Unlike the U.S., our market environment is not free from WiFi interference even on IEEE 802.15.4 channel 26. The interference problem is difficult to overcome by simply identifying alternative routes since the impact of other radios is prevalent in most cases.

Furthermore, we experimentally verify that RPL over LPL cannot provide reliability nor fair energy consumption when supporting downward-focused traffic in such dynamic and challenging wireless link environments. While providing end-to-end IP connectivity, RPL turned out to be a less suitable protocol for applications with more downwards traffic than upwards (e.g., e-price tag updates, from the markets server PC to individual low-power price tags). Based on these findings, the rest of this dissertation presents our *MarketNet* design and verifies its performance in crowded markets.

Chapter 4

MarketNet 1.0:

Asymmetric Transmission

Power-based Network

4.1 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

multi-hop 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 multi-hop routing protocols include the collection tree protocol (CTP) [21] and the routing protocol for low power and lossy networks (RPL) [18, 68], which are widely applied in sensor network deployments [1, 69]. 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 uplink paths are used as the downlink paths.

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, our e-price tag application mostly generates downward traffic (e.g., from the gateway to sensor nodes), managing multi-hop routes at each node in dynamic wireless environments will result in a significant amount of control overhead. Moreover, multi-hop routing protocols naturally require each node to have a designated memory space to store and manage these routes [27]. Most importantly, as shown in Chapter 3, multi-hop network protocols cannot achieve reliable downward packet delivery in dynamic market environments.

In this chapter, we address such inefficiencies in multi-hop routing

protocols for applications with 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 in 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) packets toward the gateway for each received packet in a multi-hop manner. This action of multi-hop 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 *MarketNet1.0* [70], 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 *MarketNet1.0*, the destination node is not required to send ACK packets to the gateway directly. Instead, *MarketNet1.0* 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, *MarketNet1.0* successfully achieves 99.4% downwards packet reception ratio (PRR) in dynamic wireless environments. While this PRR is only a 4% increase over the RPL routing protocol, *MarketNet1.0* 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 *MarketNet1.0*; 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 three-fold:

- We consider applications which mainly generate downward traffic and quantify limitations of existing multi-hop 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 *MarketNet1.0* where the gateway delivers packets to

sensor nodes via single-hop transmissions.

- Through mathematical analysis, simulations, and indoor testbed experiments, we evaluate *MarketNet1.0* against existing multi-hop routing protocols extensively, considering various channel and traffic conditions. By exploiting high transmission power of the gateway, *MarketNet1.0* can lower transmission overhead of other low power nodes by 50%.

The rest of this chapter is structured as follows. In Section 4.2 we introduce the concept of APN. We describe *MarketNet1.0* design in Section 4.3 and provide its analytical evaluations in Section 4.4. Section 4.5 shows our simulation results, Section 4.6 presents the results obtained from the indoor testbed implementation, and Section 4.7 presents the performance in an urban marketplace. We summarize with a discussion of our results in Section 4.8.

4.2 Applicability of Asymmetric Transmission Power Networks

Before discussing the details of *MarketNet1.0*, we start with introducing the APN architecture. This work targets at 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 het-

erogeneity between the gateway and common sensor nodes requires changes in the protocol design.

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

$$p_r = \frac{P}{PL(d)}. \quad (4.1)$$

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

$$PER(\gamma) = 1 - (1 - BER(\gamma))^B \quad (4.2)$$

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 an LLN, which mainly considers low power communications. This requires checking whether low-power transceivers, widely used in LLNs (e.g., CC2420 [58]), 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

$$p_{r,th} \leq p_r \leq p_{sat}. \quad (4.3)$$

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

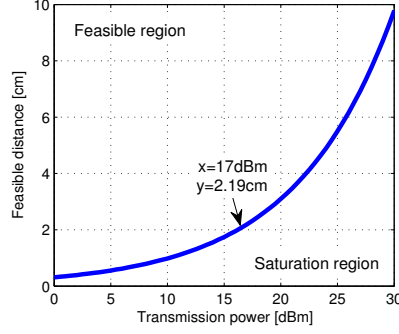
$$R_{fs} = \min \{d | p_r \leq p_{sat}\}, \quad (4.4)$$

$$R_{tx} = \max \{d | p_r \geq p_{r,th}\}. \quad (4.5)$$

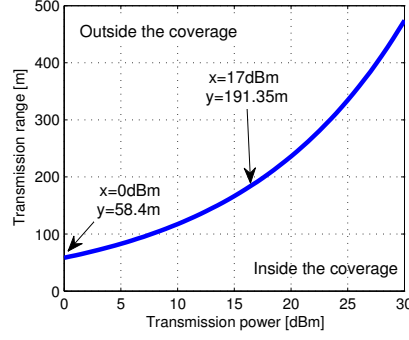
Figure 4.1 illustrates R_{fs} and R_{tx} according to the transmission power P . In this example, we assume that $p_{sat}=10\text{dBm}$ following the datasheet of CC2420 [58]. The values of $e_{th} = 0.1$ and $p_{r,th}=-87\text{dBm}$ have been confirmed through extensive experiments using CC2420 [71, 72]. In addition, we assume that $PL(d)$ follows the indoor path loss model of the IEEE 802.15.4 standard [73] given by

$$PL(d) = \begin{cases} 10^{4.02+2\log(d)}, & \text{for } d \leq 8 \\ 10^{5.85+3.3\log(d/8)}, & \text{otherwise.} \end{cases} \quad (4.6)$$

We consider transmission power in the range from 0 dBm to 30 dBm in accordance with the maximum transmission power of CC2420 [58]



(a) Feasible distance according to the transmission power



(b) Transmission range according to the transmission power

Figure 4.1: Enabling factor for an APN.

and the FCC regulations for the 2.4 GHz band [74].

Figure 4.1(a) shows that the feasible distance is negligibly small since it is shorter than 10 cm even when the maximum transmission power is applied. Figure 4.1(b) shows that the transmission range of the gateway which uses the Wi-Fi transmission power of 17 dBm [75] (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 single hop, while not power saturating the bandwidth for low-power transceivers. Overall, this result validates the applicability of APNs in low-power LLN 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.

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 multi-hop networks. Next we introduce our *MarketNet1.0*, which aims to provide an efficient best-effort packet delivery for APN architectures.

4.3 *MarketNet1.0* System Design

4.3.1 Design Overview

We design *MarketNet1.0* considering application requirements as depicted in Figure 4.2. First of all, we exploit APN architecture to well support downward-centric traffic pattern. Thus, our approach to deliver downward-focused traffic is not to improve downward routing but eliminate it. Given that the root node is typically a wall-powered device and free from energy constraint, we allow the root node to use much higher transmission power than e-price tag nodes and cover the whole area in a single hop. To achieve more reliable downward

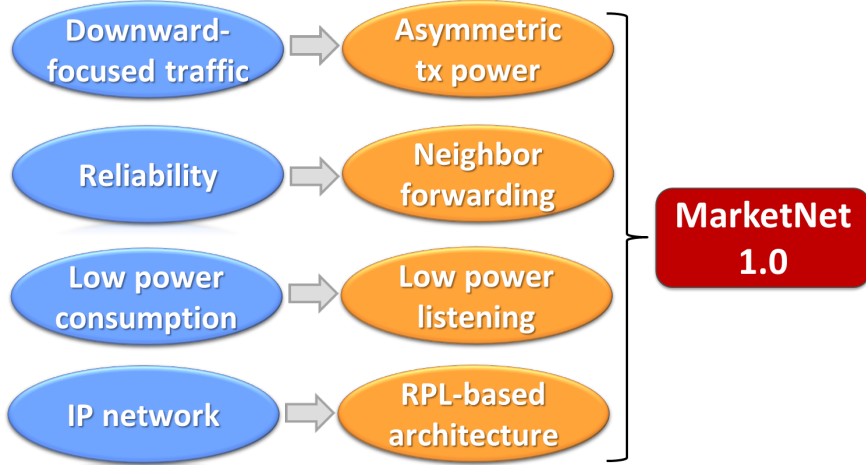


Figure 4.2: Design elements of *MarketNet1.0*.

packet delivery in dynamic market environments, *MarketNet1.0* includes neighbor forwarding mechanism which provide local retransmission when the high power root transmissions are failed. Furthermore, we exploit LPL [65] and low transmission power for e-price tags to maintain low-power profile. Lastly, we provide multi-hop uplink transmission based on RPL [18] to provide bidirectional end-to-end IP connectivity.

In *MarketNet1.0*, each node (except the gateway) is responsible for the local ACK, neighbor forwarding, and forwarding contention mitigation processes. In a multi-hop 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 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, *MarketNet1.0* prevents the network from making multi-hop downwards transmissions (with a high-power root) and also prevents ACK packets from traveling over multiple upward hops to reach the gateway.

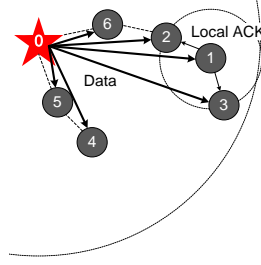
4.3.2 Neighbor Forwarding over LPL

The core design of *MarketNet1.0* is to enable neighbor forwarding over LPL. This section presents the design of neighbor forwarding mechanism design in detail. Since we described LPL characteristics in Chapter 3, we omit it in this chapter.

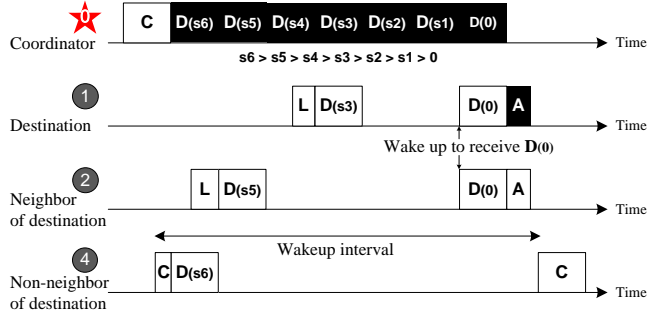
Gateway's Direct Transmissions

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, LPL 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 4.3 and 4.4 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 sleep interval. Of course, due to the asymmetric nature of APNs, the gateway does not expect to receive an ACK from the destination.



(a) Network topology



(b) Time operation

Figure 4.3: An example of direct transmission in *MarketNet1.0*. 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.

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 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 LPL. This timer runs for the sleep interval t_{sleep} . 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., $t_{bo} = 0$).

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 4.3 and 4.4), 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., $s = 0$)¹. This sleep interval can be represented as,

$$t'_{sleep}(s) = s - t_{packet}. \quad (4.7)$$

Upon receiving the $s = 0$ packet, the destination broadcasts an

¹A small guard time is added in real-implementations due to practical issues such as clock drift and per-node processing delays [76].

ACK for its neighbors to overhear. Thus, the main difference between our proposal and the baseline LPL is that the destination waits until it receives the latest of the sender's repetitive transmissions before issuing an ACK packet².

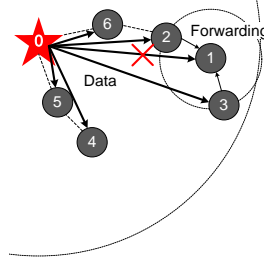
If, in the case, a node is not the packet's intended destination, the node checks whether the destination of the packet is one of its neighbors. If so, the node operates identically as the destination node up to the point when it receives packet $s = 0$. Once receiving the last packet, the node listens to the channel to overhear the ACK from the destination and goes to sleep once receiving the ACK (nodes 2 and 3 in Figure 4.3).

In the case, in which the node is neither the destination nor one of the destination's neighbors, the node sleeps its radio for the entire sleep interval (e.g., nodes 4-6 in Figure 4.3). Therefore, these nodes conserve their energy in the mean time.

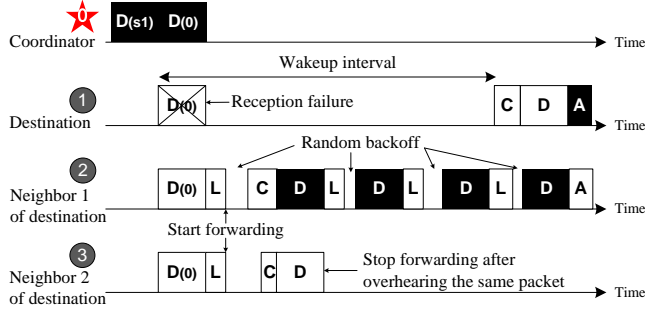
Neighbor Forwarding

As we show in Figure 4.4, 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

²With 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 to sleep mode as a way of conserving power.



(a) Network topology



(b) Time operation

Figure 4.4: An example of neighbor forwarding in *MarketNet1.0*. 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.

normal transmission of our baseline link layer protocol. That is, a neighbor node forwards the packet repetitively using a random back-off 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 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 [77]. 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.

4.4 Mathematical Performance Analysis

In this section, we analyze the performance of *MarketNet1.0* when delivering downward traffic, 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 which 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 *MarketNet1.0*, 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_{sleep} , 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 sleep 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 multi-hop down-link protocol (MHDP). In MHDP, the gateway is assumed to know a downward (multi-hop) 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 multi-hop routing protocols. We compare *MarketNet1.0* against this optimal MHDP.

4.4.1 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 retransmis-

sions, 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

$$PRR_{MHDP}(k) = PRR_{one}^{d_{hop}(k)} \quad (4.8)$$

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

Unlike MHDP, all the nodes in *MarketNet1.0* are within one hop from the gateway. Thus the PRR in *MarketNet1.0* is the same for all the nodes which is given by

$$PRR_{MNet-v1} = PRR_{direct} + PRR_{indirect}. \quad (4.9)$$

PRR_{direct} is the probability that a packet from the gateway is delivered in one hop transmission, i.e., $PRR_{direct} = 1 - e$. $PRR_{indirect}$ 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

$$PRR_{indirect} = e \left(1 - \left(1 - (1 - e)^2 \right)^{N_{ne}} \right) PRR_{one}. \quad (4.10)$$

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

$$PRR_{indirect} \geq e(1-e)^2(1-e^{n_{tx}}). \quad (4.11)$$

This shows that *MarketNet1.0* can achieve reliable packet delivery with high probability through local ACK exchange and neighbor forwarding without the gateway's confirmation.

4.4.2 Latency

A successful packet transmission incurs average latency $t_{sleep}/2$ to receive an ACK successfully which increases by t_{sleep} whenever a retransmission happens. If a packet is delivered after having j transmissions, the latency is given as $t_{sleep}(j-0.5)$. 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

$$L_{one} = \frac{t_{sleep}(1-e) \sum_{j=1}^{n_{tx}} (j - \frac{1}{2}) e^{j-1}}{PRR_{one}}. \quad (4.12)$$

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

$$L_{MHDP}(k) = d_{hop}(k) L_{one}. \quad (4.13)$$

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

In *MarketNet1.0*, the latency is the same for all the nodes. That is,

$$L_{MNet-v1} = \frac{PRR_{direct}L_{direct} + PRR_{indirect}L_{indirect}}{PRR_{MNet-v1}}. \quad (4.14)$$

Here L_{direct} is the latency when the destination node succeeds in receiving the direct transmission, denoted as $L_{direct} = t_{sleep}/2$, and $L_{indirect}$ is the latency from the destination receiving the packet via neighbor forwarding. Furthermore, $L_{indirect} = L_{one} + t_{sleep}$, where L_{one} is the forwarding latency. We can notice here that $L_{MNet-v1}$ linearly increases with t_{sleep} . In addition, with a high PRR_{direct} , *MarketNet1.0* 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.

4.4.3 Power Consumption

We analyze the basic characteristics of the baseline protocol first, followed by the power consumption performance of MHDP and *MarketNet1.0*.

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 LPL periodically wakes up and consumes energy ε_{cs} for channel sensing which is

$$\varepsilon_{cs} = p_{cs} (t_{ack} + t_{bo,max}). \quad (4.15)$$

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

$$p_{cs,tot} = \frac{N_{tot} \varepsilon_{cs}}{t_{sleep}}. \quad (4.16)$$

$p_{cs,tot}$ 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 $n_{tx,f}$, are given by

$$n_{tx,s} = \left\lceil \frac{t_{sleep}}{2(t_{packet} + t_{ack} + t_{bo,max}/2)} \right\rceil, \quad (4.17)$$

$$n_{tx,f} = \left\lceil \frac{t_{sleep}}{t_{packet} + t_{ack} + t_{bo,max}/2} \right\rceil \quad (4.18)$$

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

$$\varepsilon_{tx,s} = \varepsilon_{cs} + n_{tx,s}\varepsilon_{tx}, \quad (4.19)$$

$$\varepsilon_{tx,f} = \varepsilon_{cs} + n_{tx,f}\varepsilon_{tx}. \quad (4.20)$$

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

$$\varepsilon_{tx} = \frac{p_{idle}t_{bo,max}}{2} + p_{tx}t_{packet} + p_{rx}t_{ack}, \quad (4.21)$$

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

$$\varepsilon_{tx,one} = (1 - e) \sum_{i=0}^{n_{tx}-1} (\varepsilon_{tx,s} + i\varepsilon_{tx,f}) e^i + n_{tx}\varepsilon_{tx,f}e^{n_{tx}}. \quad (4.22)$$

$\varepsilon_{tx,one}$ increases with t_{sleep} 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

$$t_{listen} = \frac{(t_{packet} + t_{ack})(t_{bo,max} + t_{packet}/2 + t_{ack}/2)}{t_{bo,max} + t_{packet} + t_{ack}}. \quad (4.23)$$

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

$$\begin{aligned} \varepsilon_{rx,one} = & p_{tx}t_{ack}(1 - e^{n_{tx}}) \\ & + p_{rx}(t_{packet} + t_{listen}) \left\{ (1 - e) \sum_{i=1}^{n_{tx}} i e^{i-1} + n_{tx} e^{n_{tx}} \right\}. \end{aligned} \quad (4.24)$$

If a sender transmits a packet, its neighbor nodes overhear it. Since a successful packet transmission lasts for the period of $\frac{t_{sleep}}{2}$ 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 $\varepsilon_{oh,f}$, can be given as

$$\varepsilon_{oh,s} = \frac{p_{rx}N_{ne}(t_{packet} + t_{listen})}{2}, \quad (4.25)$$

$$\varepsilon_{oh,f} = p_{rx}N_{ne}(t_{packet} + t_{listen}). \quad (4.26)$$

The total energy consumption for overhearing a one-hop transmission

becomes

$$\varepsilon_{oh,one} = (1 - e) \sum_{i=0}^{n_{tx}-1} (\varepsilon_{oh,s} + i\varepsilon_{oh,f}) e^i + n_{tx}\varepsilon_{oh,f}e^{n_{tx}}. \quad (4.27)$$

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 $E_{MHDP}^{tx}(k)$ and $E_{MHDP}^{rx}(k)$, can be expressed as

$$E_{MHDP}^{tx}(k) = \varepsilon_{tx,one} \sum_{i=1}^{N_{hop}(k)-1} (1 - e^{n_{tx}})^i, \quad (4.28)$$

$$E_{MHDP}^{rx}(k) = \varepsilon_{rx,one} \sum_{i=0}^{N_{hop}(k)-1} (1 - e^{n_{tx}})^i. \quad (4.29)$$

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

$$E_{MHDP}^{oh}(k) = \varepsilon_{oh,one} \sum_{i=0}^{N_{hop}(k)-1} (1 - e^{n_{tx}})^i. \quad (4.30)$$

Obviously, $E_{MHDP}^{tx}(k)$, $E_{MHDP}^{rx}(k)$, and $E_{MHDP}^{oh}(k)$ 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 heavy burden imposed by relaying.

Let us denote the packet inter arrival 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

$$P_{MHDP} \tag{4.31}$$

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

$\varepsilon_{tx,one}$ in $E_{MHDP}^{tx}(k)$ increases, $p_{cs,tot}$ decreases with t_{sleep} , 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 LLNs efficiently.

MarketNet1.0 Analysis

In *MarketNet1.0*, the energy consumption of all non-gateway nodes are 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 fi-

nally 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

$$E_{MNet-v1}^{tx} \quad (4.32)$$

$$= \left(1 - (1 - e)^2\right) \left\{ \left(1 - \left(1 - (1 - e)^2\right)^{N_{ne}}\right) \varepsilon_{tx,one} + \left((1 - e)^2 N_{ne} - 1\right) \left(\frac{p_{idle} t_{bo,max}}{2} + \varepsilon_{cs}\right) \right\}.$$

The first term in brackets represents the energy consumption at the forwarding node that occupies the medium first, and similarly the second term are 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

$$E_{MNet-v1}^{rx} = \frac{3p_{rx}t_{packet}}{2} + (1-e)p_{rx}t_{packet} + (1-e)^2p_{tx}t_{ack} \quad (4.33) \\ + \left(1 - (1-e)^2\right) \left(1 - \left(1 - (1-e)^2\right)^{N_{ne}}\right) \varepsilon_{rx,one}.$$

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 *MarketNet1.0*, 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

$$E_{MNet-v1}^{oh} = \frac{3p_{rx}t_{packet}N_{tot}}{2} + (1-e)N_{ne}p_{rx}(t_{packet} + t_{ack}) \quad (4.34) \\ + \left(1 - (1-e)^2\right) \left(1 - \left(1 - (1-e)^2\right)^{N_{ne}}\right) \varepsilon_{oh,one}.$$

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 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 *MarketNet1.0* can

be written as

$$P_{MNet-v1} = p_{cs,tot} + \frac{E_{MNet-v1}^{tx} + E_{MNet-v1}^{rx} + E_{MNet-v1}^{oh}}{t_{int}}. \quad (4.35)$$

As in the analysis of MHDP energy consumption, $P_{MNet-v1}$ is directly related with t_{sleep} due to $p_{cs,tot}$ and $\varepsilon_{tx,one}$. The weight for $\varepsilon_{tx,one}$ in *MarketNet1.0* is lower than that of MHDP, since *MarketNet1.0* works independently from the number of nodes and its relaying occurs only when the gateway's transmission fails. The low transmission overhead allows *MarketNet1.0* to consume much lower energy with a large sleep interval, especially in large scale LLNs, compared to other multi-hop protocols.

4.5 Simulation Results

Based on our mathematical analysis, we now move on to performance evaluation of *MarketNet1.0* and MHDP through simulation. Specifically, we developed an event-driven simulator that considers real channel environments according to the path loss model of Eq. (4.6) which is suitable for indoor environments [78]. 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 [70]. We assume that each low-power node uses a transmission power of 0 dBm

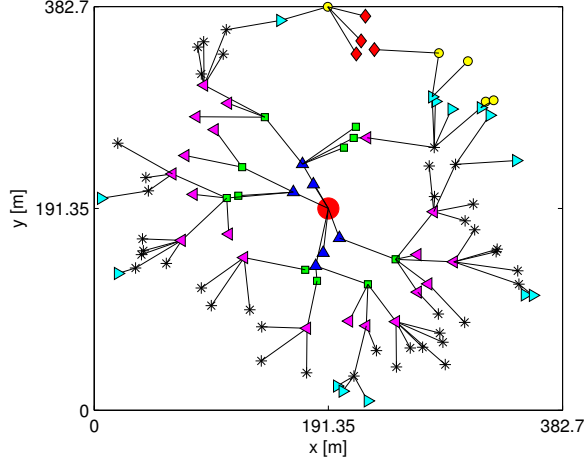


Figure 4.5: Simulation network topology of 100 nodes.

(i.e., the maximum transmission power of CC2420), while the gateway in *MarketNet1.0* uses 17 dBm as a WiFi transceiver [75]. Our feasibility analysis presented in Section 4.2 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 4.5, 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 [73], we obtain the BER as

$$BER(\gamma) = \frac{8}{15} \times \frac{1}{16} \times \sum_{k=2}^{16} (-1)^k \binom{16}{k} \exp \left(20 \left(\frac{1}{k} - 1 \right) \gamma \right). \quad (4.36)$$

We set $p_{tx}=70\text{mW}$ ³, $p_{rx}=78\text{mW}$, $p_{cs}=30\text{mW}$, and $p_{idle}=3.7\text{mW}$ following the CC2420 specification [58].

We assume that the channel sensing range is the same as the transmission range, 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 sec. Each simulation runs for 4 hours.

In Figure 4.5, 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.

4.5.1 Latency

Figure 4.6 plots average packet delivery latency observed by varying the sleep interval. Here, the horizontally dotted red line indicates the inter-packet-interval (IPI) of 15 sec. As expected, infrequent wakeup naturally results in increased delivery latency. *MarketNet1.0* achieves

³ p_{tx} is the *total* power consumption of a node in transmit mode.

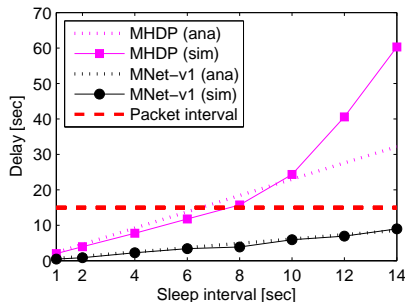


Figure 4.6: Latency vs. sleep interval.

significantly lower latency than MHDP as sleep interval increases. This is mainly due to the fact that in *MarketNet1.0*, 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 sleep 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.

4.5.2 Packet Delivery Ratio

Packet delivery ratio is an important metric in many applications, which is shown in Table 4.1. Although the analytical results show that MHDP is on par with *MarketNet1.0* in packet delivery performance, our simulation results differ a bit from those of analysis. Deep

	2s	6s	10s	14s
MHDP (ana.)	10^{-5}	10^{-5}	10^{-5}	10^{-5}
MHDP (sim.)	0	0	0	$3 \cdot 10^{-2}$
MNet-v1 (ana.)	10^{-5}	10^{-5}	10^{-5}	10^{-5}
MNet-v1 (sim.)	0	0	0	0

Table 4.1: Packet delivery failure ratio vs. sleep interval

investigation on the simulation traces reveal that packet loss in MHDP is higher than that in *MarketNet1.0* when the sleep interval is large, mainly due to the queueing loss. The queueing delay in MHDP results in high packet latency as presented in Figure 4.6. Therefore, buffer provisioning in MHDP is an important concern that *MarketNet1.0* avoids by transmitting downward packets in one hop.

4.5.3 Power Consumption

In Figure 4.7, we show the average power consumption per node with respect to the sleep interval. The power consumption of MHDP first decreases and then increases with the sleep interval. This is caused by the increased transmission burden from the sleep interval. In contrast, *MarketNet1.0* consumes less power than MHDP, and its power consumption continuously decreases with the sleep interval.

We break down the power consumption factors in detail using Figure 4.8. As expected, power consumption for channel sensing in MHDP and *MarketNet1.0* decreases with the sleep interval. Regarding other power consumption factors, the transmission power consumption in MHDP increases with the sleep interval due to prolonged

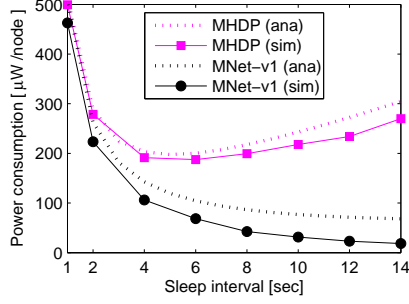


Figure 4.7: Power consumption vs. sleep interval.

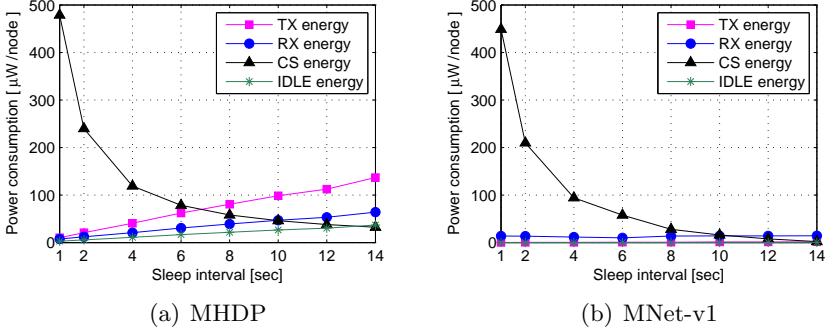


Figure 4.8: Detailed power consumption breakdown.

repetitive transmission. On the other hand, since *MarketNet1.0* finishes its transmission mostly via single-hop (from the gateway), the power consumption of the low-power nodes remain very low.

Lastly, we present the fairness in power consumption using the Jain's Fairness Index [70]

$$J(p) = \frac{\left(\sum_{i=1}^{N_{tot}} p_i \right)^2}{N_{tot} \sum_{i=1}^{N_{tot}} p_i^2}, \quad (4.37)$$

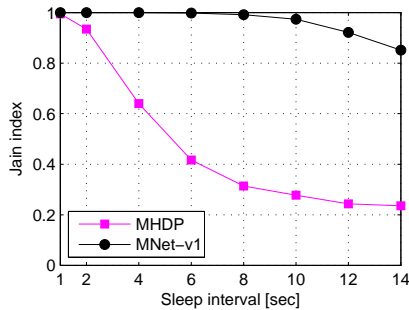


Figure 4.9: Fairness in power consumption vs. sleep interval.

where p_i is the power consumption of node i . In Figure 4.9, *MarketNet1.0* shows an overall fair performance compared to MHDP. This is mainly because in a multi-hop 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 [34]. In *MarketNet1.0*, fair power consumption can be achieved since the gateway covers the whole network directly.

4.6 Testbed Experiments

4.6.1 Implementation and Environment Setting

We implemented *MarketNet1.0* on top of RPL and LPL in TinyOS [18, 65]. Unlike the ideal (and unrealistic) MHDP, RPL requires control overhead in finding routes and suffers from limited or outdated routing information. Figure 4.10 summarizes some of the major modifications made to the software stack of TinyOS. Specifically, we removed the downlink route discovery procedures and downward routing table in

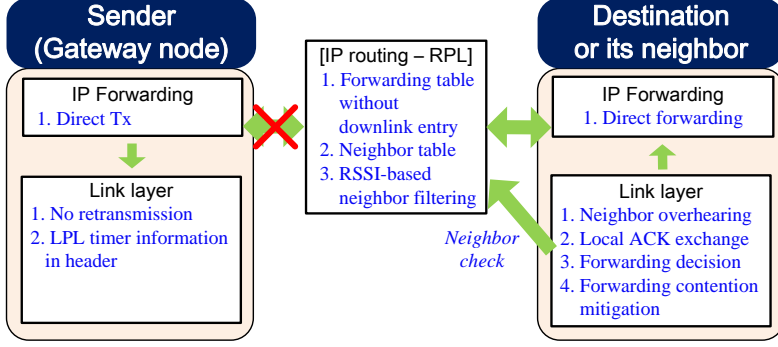


Figure 4.10: *MarketNet1.0* architecture.

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 *MarketNet1.0* uses neighbor information for local ACK exchange and neighbor forwarding. If a node with an inconsistent link is added to the neighbor table, *MarketNet1.0* 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 pre-defined 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. *MarketNet1.0* 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.

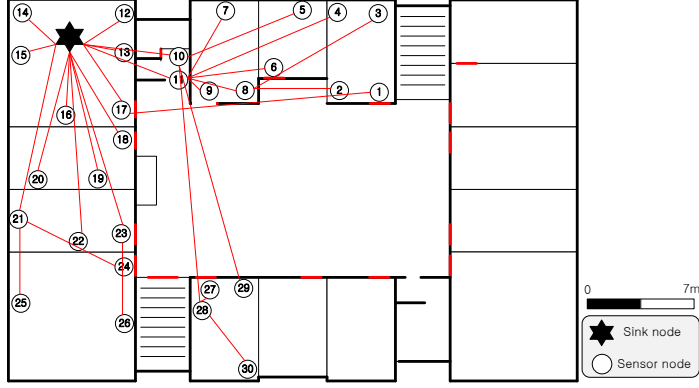


Figure 4.11: Testbed topology map.



Figure 4.12: Left: MTM-CM3300MSP for the high-power gateway, Right: K mote for the low-power nodes.

Figure 4.11 shows the testbed topology map consisting of 20 nodes and one gateway, marked with a star, in an office environment. For a low power node, we use the same device as in Chapter 3. For the high powered gateway, we use MTM-CM3300MSP [79], which is similar to TelosB [56] but includes a 10 dB power amplifier as in Figure 4.12. In our experiments, the high power gateway and common low-power nodes use transmission power 10 dBm and -15 dBm, respectively, with an antenna gain of 5 dB. To maintain both multi-hop uplink

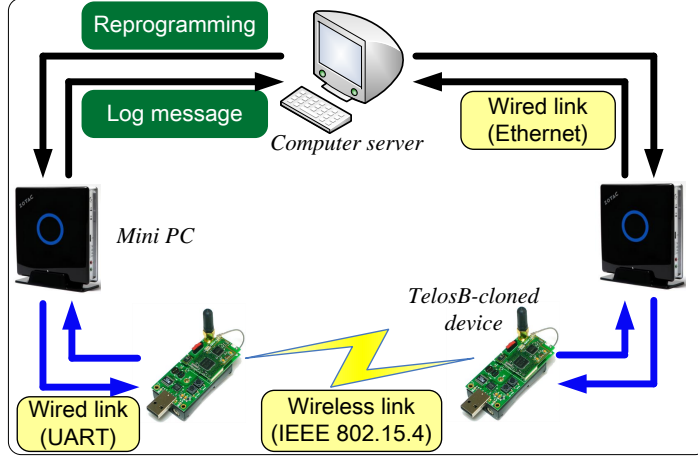


Figure 4.13: Testbed architecture.

routes and single hop downlink connectivity, the high power gateway transmits a data packet with 10 dBm and other packets (e.g., routing packets and ACKs) with -15 dBm. Given this testbed configuration, the RPL implementation connects all the nodes in a maximum of three hops. From empirical results, we set the threshold $p_{r,th}$ as -87 dBm when deciding to include a node into the neighbor table. Finally, each node has a FIFO buffer size of 10 packets.

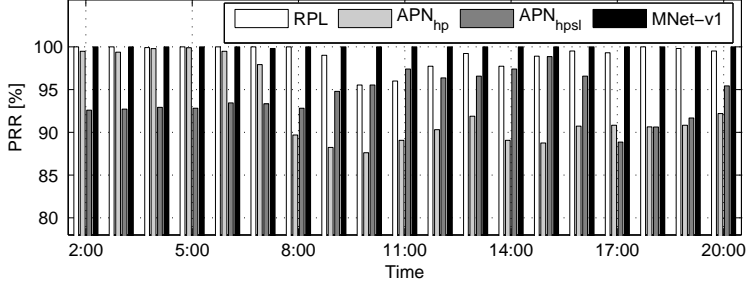
As we illustrate in Figure 4.13, each low power node is connected to a PC via USB and sends log messages to the PC through the UART back-channel. We gather the log messages from each PC through ethernet to obtain various performance measurements and real-time operation statuses. Furthermore, we remotely 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.

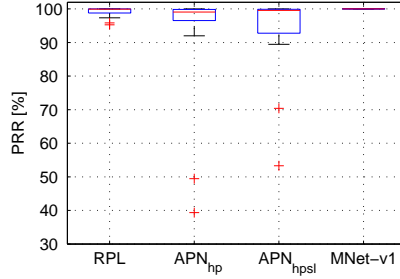
4.6.2 Downward Traffic Scenario

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 sec (i.e., 3.75 sec from the network’s perspective), and set the asynchronous sleep interval of each node to 0.5 sec. To verify the effectiveness of each design element in *MarketNet1.0*, we also evaluate the performance of two variants of *MarketNet1.0*, 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 *MarketNet1.0*.

In Figure 4.14(a), we present the per-hour average PRR throughput per day. During the day time, the PRR performance of RPL



(a) Per-hour PRR performance



(b) Per-node PRR performance

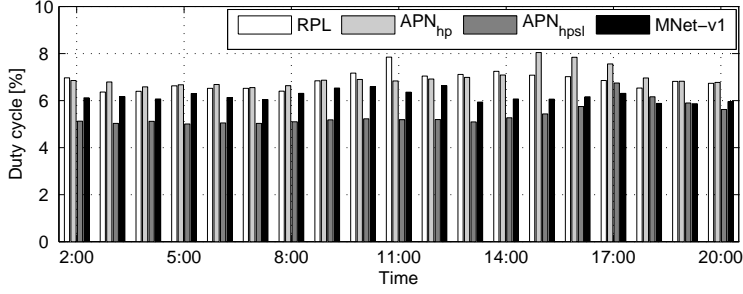
Figure 4.14: PRR performance of RPL, APN_{hp}, APN_{hpsl}, and *MarketNet1.0* for 19 hours (from 2AM to 9PM), where APN_{hp} and APN_{hpsl} are variants of *MarketNet1.0*.

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 *MarketNet1.0*, we always observe not only the most stable but also highest PRR performance among the competitive protocols mainly owing to the com-

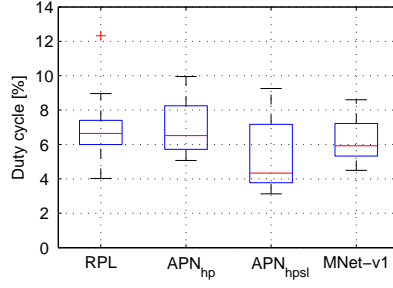
combination of high-power single-hop transmissions at the gateway and local retransmissions at neighbor nodes of the packet’s destination. This implies that the link instability due to channel dynamics significantly impacts the performance of both multi-hop and high-powered single-hop communications while this is not an issue in *MarketNet1.0*.

Figure 4.14(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 4.11, 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, *MarketNet1.0* 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 *MarketNet1.0* is beneficial to maximizing the strength of APNs and leads to reliable packet delivery.

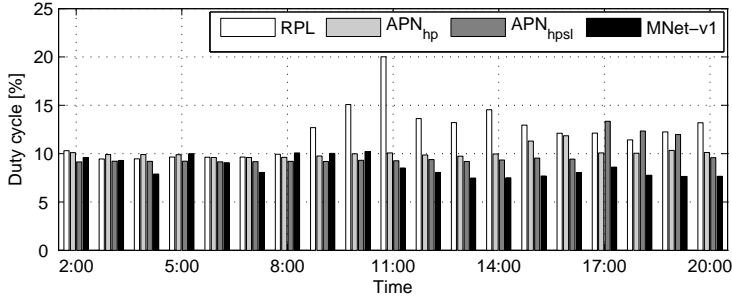
We further focus on the duty cycle performance in Figures 4.15(a)-4.15(c), which indirectly represents the nodes’ energy consumption. First, Figure 4.15(b) shows that RPL’s duty cycle performance is significantly unfair among nodes. The unbalanced transmission overhead among nodes in RPL is unavoidable due to its multi-hop nature since nodes near the gateway are asked to relay more packets compared



(a) Average per-hour duty cycle performance



(b) Per-node duty cycle performance



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

Figure 4.15: Duty cycle performance of RPL, APN_{hp} , APN_{hpsl} , and *MarketNet1.0* for 19 hours (from 2AM to 9PM), where APN_{hp} and APN_{hpsl} are variants of *MarketNet1.0*.

to close-to-leaf nodes. Furthermore, RPL inherits the load balancing problem [34], which worsens the unfair energy consumption among nodes. From Figures 4.15(a) and 4.15(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 4.14(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 *MarketNet1.0* is slightly lower than that of RPL, and for the worst case, the differences become far more prominent. This is because *MarketNet1.0* provides low and fair transmission overhead for nodes by allowing the gateway to take most of transmission burden. Quantitatively, *MarketNet1.0*’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} , *MarketNet1.0* provides only slightly higher radio duty-cycles due to the neighbor forwarding overhead. Given other performance metrics, we find this

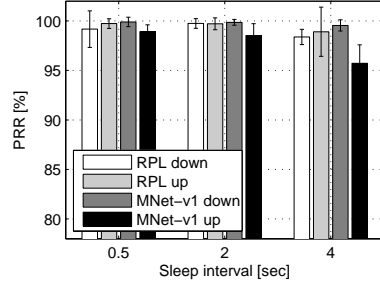
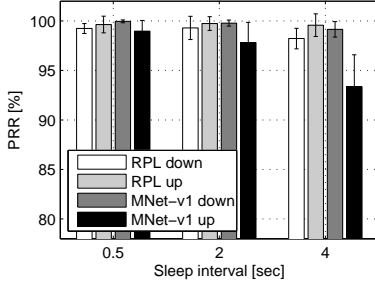
as a reasonable cost to pay.

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. *MarketNet1.0* successfully takes the advantages of an APN architecture and significantly improves the networking performance over RPL.

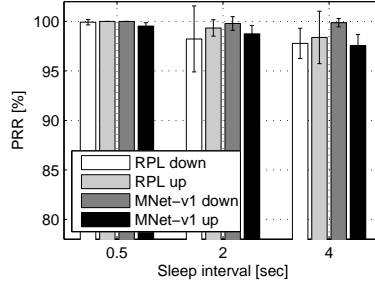
4.6.3 Mixed Traffic Scenario

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

In Figure 4.16, 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 *MarketNet1.0* is superior to that of RPL in all cases. For upward traffic, *MarketNet1.0* is on par with RPL for short sleep intervals and shows slightly decreased performance with the sleep interval. This is mainly because high power downward broadcast packets can be queued due to infre-



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



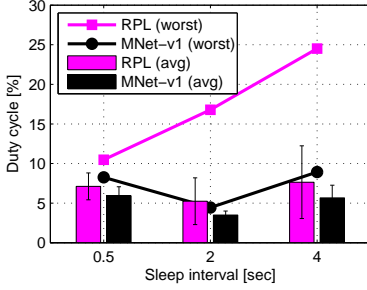
(c) Downlink interval = 300s, uplink interval = 100s

Figure 4.16: Average PRR performance vs. sleep interval.

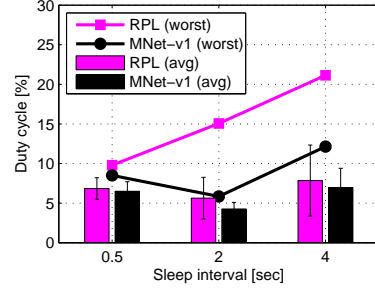
quent wakeups and this congestion causes packet loss at low power nodes. *MarketNet1.0* provides nearly perfect downward PRR regardless of traffic patterns, but its upward PRR decreases as the downward traffic becomes dominant. Each high-powered transmission in *MarketNet1.0* incurs wireless interference throughout the whole network, which causes frequent packet collisions at low power nodes when the network generates heavy downward traffic⁴.

In Figure 4.17, we note an interesting observation for the duty

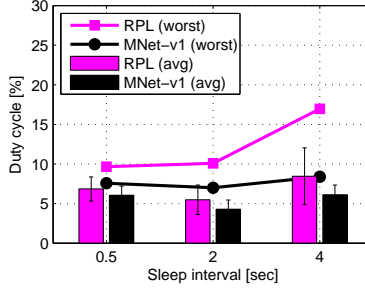
⁴We consider improving upward packet delivery performance in APNs as part of our future work.



(a) Downlink interval = 100s, uplink interval = 300s



(b) Downlink interval = 150s, uplink interval = 150s



(c) Downlink interval = 300s, uplink interval = 100s

Figure 4.17: Duty cycle performance vs. sleep interval.

cycle of each node in both the average case and worst case. For all the nine instances, the average duty cycle of *MarketNet1.0* is lower than that of RPL. The duty cycle observed from the worst case node in *MarketNet1.0* is significantly lower compared to that of RPL. This implies that *MarketNet1.0* has a significantly longer network lifetime than RPL since the battery lifetime of the first dead node impacts the usability of the entire system.

Next we turn our attention to the per-hour performance of the test network. For this, we take the traffic case of [100 sec, 300 sec] with

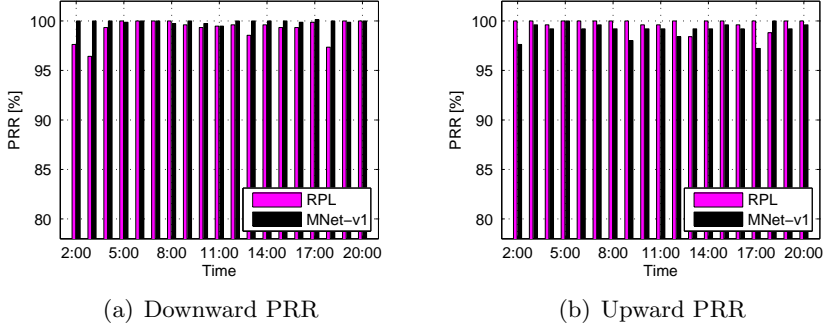


Figure 4.18: Per-hour average PRR performance for downward and upward traffic.

the sleep interval of 0.5 sec and plot the PRR performance for each hour in Figure 4.18. In Figure 4.18(a), *MarketNet1.0* shows steady downward performance over time. This result implies that the high power transmission and local retransmission features of *MarketNet1.0* help to achieve steady packet delivery performance despite the varying channel condition throughout the day. For upward PRR performance shown in Figure 4.18(b), RPL shows slightly higher PRR than *MarketNet1.0*. This is because RPL is optimized for data collection, and downward routes are simply set as the reverse of upward routes.

Figure 4.19 shows the overhead for each packet transmission. The overhead in *MarketNet1.0* 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 *MarketNet1.0* does

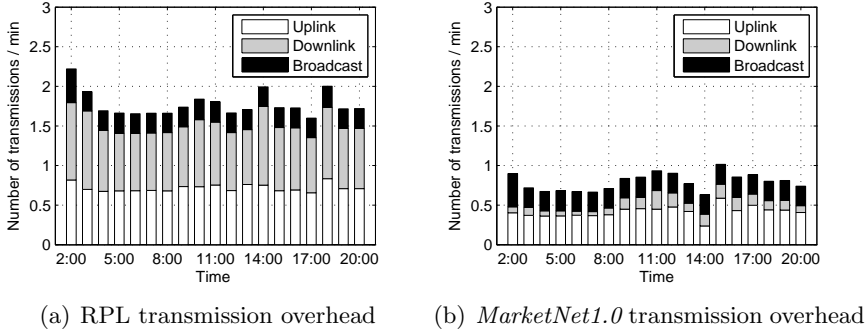


Figure 4.19: Per-hour average transmission overhead.

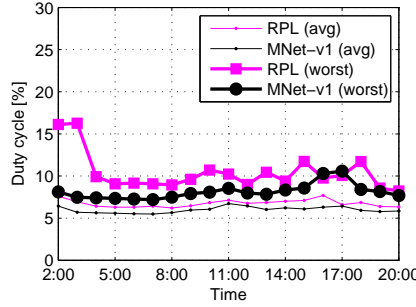


Figure 4.20: Per-hour duty cycle performance.

not generate such control packets.

Lastly we observe the average per-hour duty cycle of each node and that of the worst case node over time in Figure 4.20. When combining the results of Figure 4.15 and Figure 4.20, we observe that each node in *MarketNet1.0* has a lower duty-cycle compared to RPL under various traffic scenarios. Overall, our experimental results show that in a real-world test environment *MarketNet1.0* outperforms the multi-hop routing protocol of RPL when reliable and energy efficient delivery of downward packets is considered.

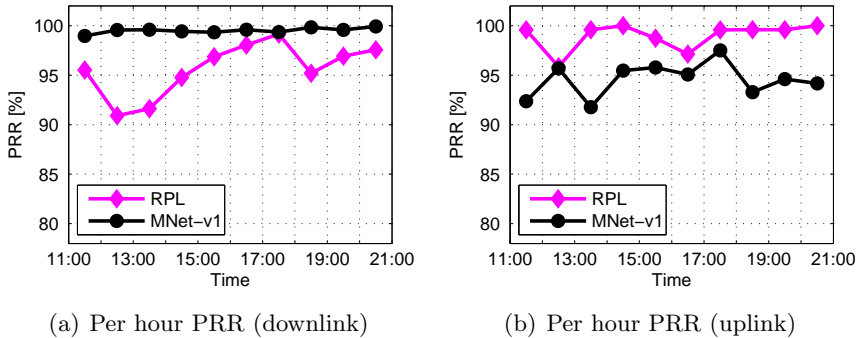
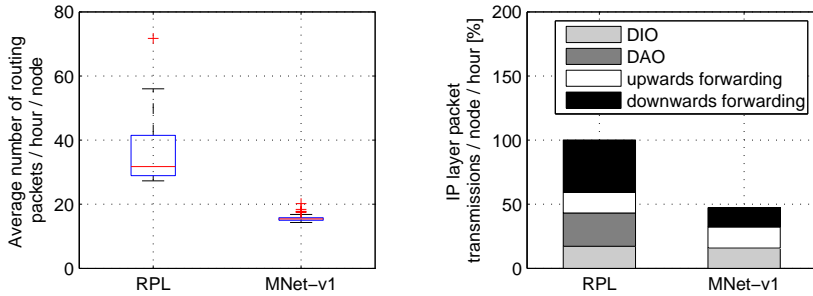


Figure 4.21: Packet loss ratio of RPL and *MarketNet1.0* for uplink and downlink traffic. *MarketNet1.0* shows a lower loss rate for downlink traffic, while RPL outperforms on uplink packet delivery performance.

4.7 Market Deployment

For evaluating *MarketNet1.0* in the market, we deploy nodes identically to our preliminary measurements in Chapter 3 (c.f., Figure 3.11). Furthermore, we generate the same traffic pattern and select the same set of networking parameters as in the preliminary measurements (e.g. LPL sleep interval of 2 seconds). While experiments were not performed simultaneously (i.e., 11AM-9PM on different days), which could lead to inconsistent results due to potential interference from other systems and differences in WiFi activities over different wireless channels, we confirmed that for the three days of testing (e.g., one for each system), the number of market customers were roughly similar.

Figure 4.21 presents the packet loss trends for downlink and uplink traffic when using RPL- and *MarketNet1.0*-based networks for 10 hours (11AM - 9PM). For downlink, *MarketNet1.0* outperforms RPL



(a) Routing overhead distribution of each node (b) Average transmission overhead per node per hour

Figure 4.22: Transmission overhead of RPL and *MarketNet1.0*. *MarketNet1.0* provides lower overhead by suppressing DAO messages and multihop forwarding.

mainly due to the fact that RPL, to begin with, was not designed to provide optimized downwards routes. Rather, RPL typically achieves downwards routing using the reverse of the upwards routes, despite the asymmetry of wireless links [80]. In RPL, the routes are adjusted only by the children nodes; thus, if the parent experiences severe downlink packet loss, it takes a while for the parent to inform children of its status. Furthermore, control packet losses further delay these route updates. Thus, RPL results in high packet loss when focusing more on downwards traffic than upwards traffic in dynamic channel environments. On the other hand, *MarketNet1.0* uses higher power for downlink transmissions and local retransmissions to assure the reliability as well as to support two-hop nodes.

Nevertheless, for uplink, RPL outperforms *MarketNet1.0* despite *MarketNet1.0* constructing uplink routes using RPL. This is due to

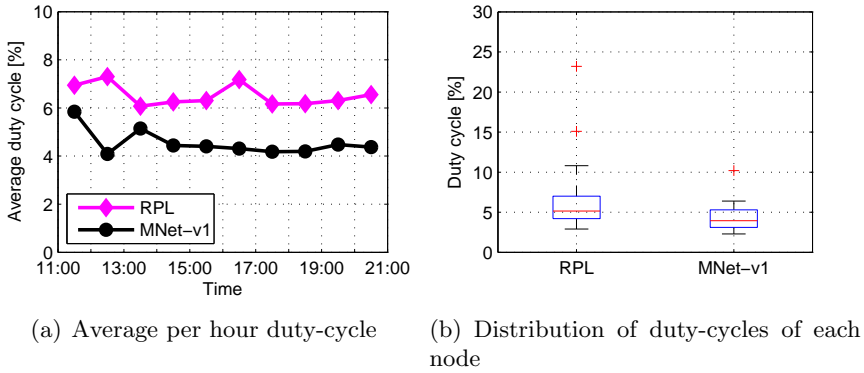


Figure 4.23: Duty-cycle of RPL and *MarketNet1.0*. *MarketNet1.0* improves duty-cycle performance due to lower transmission overhead.

the interference caused by the high-power downward transmissions in *MarketNet1.0*. Specifically, uplink packets face communication challenges from the root's downwards transmissions, leading to high *MarketNet1.0* packet loss. Nodes distant from the root are more significantly affected by this interference since they are outside the root's clear channel assessment (CCA) range. Unfortunately, this cannot be resolved by simply provisioning new upward routes since the high-power transmission covers the entire network.

Figures 4.22(a) and 4.22(b) plot the per hour routing overhead of each low-power node and their average transmission overhead at the IP layer, respectively. From Figure 4.22(a) we notice that the routing overhead of *MarketNet1.0* is approximately 52% lower than RPL. While *MarketNet1.0* uses RPL for its upwards routes, we see this reduction due to the fact that *MarketNet1.0* suppresses destination advertisement object (DAO) messages, which are periodically

issued at each non-root node to initiate downwards RPL routes. Furthermore, Figure 4.22(b), where we present a breakdown of packet transmissions, shows that *MarketNet1.0* results in lower transmission overhead by removing multi-hop forwarding for downwards transmissions and minimizing route control packets.

Finally, Figure 4.23 shows the distribution of duty-cycle of each node, including the total radio-on time not only for its transmissions but also for reception, overhearing, and LPL idle listening. It shows that *MarketNet1.0* achieves a lower duty-cycle than RPL (i.e., 30% lower on average, and 66% lower for the worst performing node), which is due to the reduction in transmission overhead for each non-root node.

Overall, our preliminary studies show that RPL- and *MarketNet1.0*-based networks each possess their own advantages and disadvantages. While RPL provides reliable upward performance, *MarketNet1.0* operates exceptionally well for downwards traffic and improves the energy consumption by reducing the transmission overhead. With these results in mind, we emphasize once again that our target application asks for effective *downwards* traffic delivery. Nevertheless, it is also important that upwards delivery is reliable, given that messages such as rack status updates and application layer acknowledgments are carried via upwards packets.

4.8 Summary

In this chapter, we present *MarketNet1.0* which comprises asymmetric transmission power-based network architecture, neighbor forwarding, low power listening, and RPL-based uplink routing. In *MarketNet1.0*'s asymmetric transmission power-based network, 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 and perform multi-hop upward transmission using RPL. *MarketNet1.0*'s neighbor forwarding comprises sub-components such as the gateways 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, empirical testbed experiments, and market deployments, we confirm that *MarketNet1.0* outperforms other competitive multi-hop routing protocols with respect to downlink PRR, radio duty cycle, and transmission overhead. On the other hand, we observed that *MarketNet1.0* provides lower uplink PRR than RPL since high power root transmissions reduce spatial reuse for low-power uplink transmissions. Although our target application mainly requires reliable downward packet delivery, improving uplink PRR is still valuable and this motivates us to do further work described in Chapter 5.

Chapter 5

MarketNet 2.0: Network-wide Superframe Architecture

5.1 Introduction

The lessons from Chapter 3 suggest that constructing a two-way multi-hop network of symmetric transmission power links (with RPL) provides ‘reasonable’ performance for our target environment, but is not ideal in terms of overhead and downwards packet delivery reliability. On the other hand, design and performance evaluation of *MarketNet1.0* in Chapter 4 shows that an APN provides satisfactory downlink performance, but fails in providing satisfactory uplink performance due to lack of spatial reuse caused by the high-power root

transmissions.

In order to compensate for the increased interference range of the root, the APN should provide an end-to-end latency significantly shorter than the one-hop latency of the homogeneous transmission power-based network (HPN). Naturally, as the work in [70] also shows, APNs reduce this latency by design. However, reducing the latency to be below that of one-hop packet transmission latency of HPN is non-trivial; thus, this suggests that we need to significantly redesign and improve *MarketNet1.0* to achieve reliable upward packet delivery.

Given these characteristics, in this chapter, we now design a more suitable solution for our application. In doing so, we take the observation that our application heavily relies on satisfactory downwards packet delivery performance (e.g., price updates). Therefore we borrow some of the concepts proposed in *MarketNet1.0* in enabling APN-based systems to design *MarketNet2.0*. Specifically, *MarketNet2.0* puts RPL at its basis in order to provide end-to-end multi-hop IPv6 routing, but it is a complete re-design of RPL with the concept of APNs that adopts the advantages of both RPL and *MarketNet1.0*. As our results will later show, *MarketNet2.0* enhances the performance over both RPL and *MarketNet1.0* in our target environment.

The rest of this chapter is structured as follows. We describe *MarketNet2.0* design in Section 5.2. Section 5.3 presents the results obtained from both testbed experiments and market deployments. We summarize with a discussion of our results in Section 5.5.

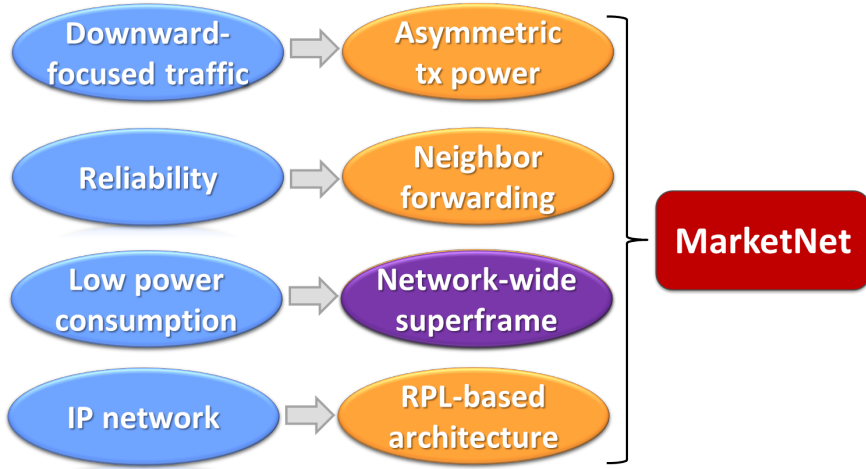


Figure 5.1: Design elements of *MarketNet2.0*.

5.2 MarketNet2.0 System Design

The design elements of *MarketNet2.0* is described in Figure 5.1, which shows that, unlike *MarketNet1.0*, *MarketNet2.0* exploits a network-wide superframe architecture as its underlying link layer, rather than LPL. Our intuition behind this design is that an APN can minimize the end-to-end latency of downward packets by using the high transmission power at the root to achieve not only single hop downwards packet delivery (as *MarketNet1.0*) but also the *network-wide time synchronization* in the time domain. Based on the time synchronization, *MarketNet2.0* allows all low-power nodes in the network net to wake up and sleep their radios simultaneously and use a single superframe (i.e., network-wide superframe).

Compared to LPL, which requires half of the sleep interval, on average, for one hop packet delivery (due to the uncertain neighbors' wake-up schedules), our superframe-based scheme significantly reduces the packet delivery latency to only a packet length while reserving more time for upward packets and radio sleep. Furthermore, the superframe dedicates and separates transmission periods for the root and individual low-power nodes using time division duplex, which protects low-power nodes from the root's high power transmission interference and provides spatial reuse. Lastly, the reduction in packet transmission time allows nodes to maintain very short radio active periods to improve their energy efficiency.

MarketNet2.0 uses the term “network-wide superframe” given that it is noticeably different from the “cluster-wide superframes” in the IEEE 802.15.4 beacon-enabled mode. Specifically, since *MarketNet2.0* uses a high transmission power root, the entire network is synchronized into a single superframe without the need for multiple disjoint cluster heads, and it requires only the root to transmit beacon messages for time synchronization. This allows individual nodes to share a single superframe without any additional overhead. Furthermore, unlike IEEE 802.15.4 beacon-enabled mode which requires microsecond accuracy in time synchronization to provide slotted CSMA (each slot length is 0.32ms), our superframe-based scheme requires only millisecond-scale accuracy to detect the start of each superframe and provides a simple way of synchronization.

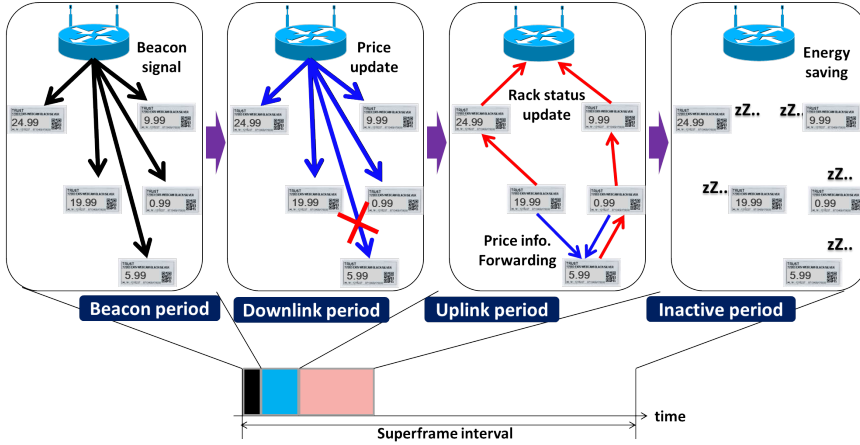


Figure 5.2: Superframe structure of *MarketNet2.0*. Downlink and uplink transmission periods are designated using a beacon message sent by the root.

5.2.1 Network-wide Superframe Architecture

This section describes *MarketNet2.0*'s superframe architecture in detail, including the basic superframe operations, uplink period partitioning, and initial synchronization.

Basic Operation

Figure 5.2 illustrates *MarketNet2.0*'s network-wide superframe structure. A superframe consists of an inactive period and an active period, where the active period is further divided into the beacon, downlink, and uplink periods. At the start of a beacon period, the root broadcasts a *regular beacon* frame with high transmission power, which contains three types of information: (1) A *beacon timer* which is the time interval until the next regular beacon transmission (i.e., start

of the next superframe), (2) next downlink period duration, and (3) next uplink period duration.

A non-root node wakes up just before the start of the agreed upon beacon period and receives the next incoming (expected) beacon frame. Using the information in the beacon, a non-root node configures its downlink, uplink and idle durations. Given that these event times are controlled centrally at a single point, a positive side effect is that the duty-cycle (and in turn the lifetime) can be easily estimated and controlled at the central server. Furthermore, although the downlink period starts directly after the beacon reception, the periodic nature of the beacon interval allows the non-root nodes to wake up accurately at a level of reasonable synchronization, despite missing several beacon messages. While many time synchronization schemes exist (e.g. FTSP [81]), we take a simple approach since our application requirements are at millisecond-level accuracy.

When nodes enter the downlink period, only the root is allowed to transmit. All other nodes simply receive packets and transmit ACKs when required in between transmissions. Our current design of *MarketNet2.0* makes multiple unicast IPv6 packet transmissions to individual nodes from the root instead of transmitting batches of downwards messages via multicast. We take such a design choice due to three reasons. First, we try to reduce the implementation complexity at the low-power non-root nodes (e.g., avoid complex multicast addressing). Second, unicast message transmission would mean that

the entire network is idle except the destination node that transmits its ACK packet for the downwards message; therefore, minimizing the contention during ACK transmissions and avoids ACK-explosion. Third, the aggregation of price update messages can cause additional latency at the central server. Nevertheless, we foresee the issue of aggregating messages and making multicast transmissions as an interesting future work.

MarketNet2.0 includes a best-effort retransmission scheme to increase the reliability of data delivery using a predefined maximum number of retransmissions. In APNs, while the root’s transmission successfully reaches the destination, the ACK cannot reach the root over single-hop. To provide reliable delivery in such cases, *MarketNet2.0* utilizes “local ACK packets” rather than end-to-end ACKs, and borrows the neighbor forwarding scheme in *MarketNet1.0* [70]. Specifically, neighbor nodes of the destination confirm the downward packet delivery on behalf of the root by overhearing both the data and ACK packets. If an ACK is not overheard, neighboring nodes locally retransmit the overheard downlink packet to the destination during the following uplink period. Once the downlink duration ends, the uplink period starts for its pre-defined duration (as specified in the beacon) so that low-power nodes can send their messages or perform neighbor forwarding. Following this, the superframe specifies an idle period for the nodes to turn off their radios until the next beacon reception.

Uplink Period Partitioning

Unlike the beacon-enabled mode in the IEEE 802.15.4 standard which requires each cluster head to have an independent superframe duration (i.e., wake-up at different times) for mitigating inter-cluster interference [73], *MarketNet2.0* allows all low-power nodes to share a single superframe since it has a root enabled to cover the entire target area using high-power transmissions. This naturally brings reduced implementation complexity and a common wake-up schedule for messages such as RPL control messages. However, such operations can cause congestion as multiple nodes compete for channel access within a limited uplink duration.

To alleviate this problem, as Figure 5.2 shows, we partition the uplink period into several sub-periods so that each is just long enough to transmit one packet assuming a maximum IEEE 802.15.4 frame length. The intuition behind this partitioning is to combine the benefits of CSMA with a TDMA-based approach. Specifically, a node that intends to send packets selects a sub-period (at random) and suppresses its packets unless they are at the beginning of this sub-period *and* its CCA succeeds. This constraint helps reduce the number of contenders on the channel [82]. Furthermore, *MarketNet2.0* uses priority-based random backoff as a function of the queue occupancy to allow congested nodes to utilize relatively more sub-periods in the uplink period.

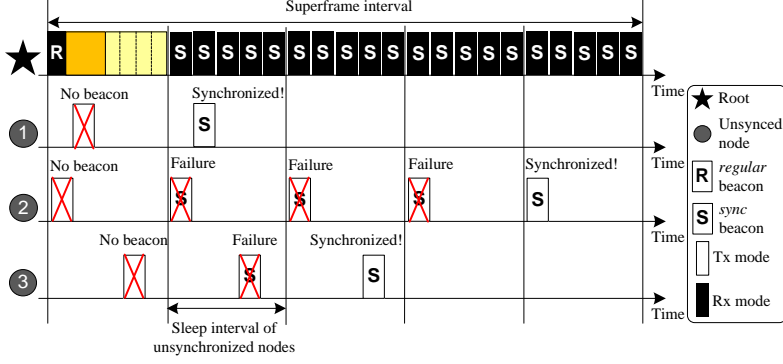


Figure 5.3: An example of *MarketNet2.0*'s initial synchronization procedure. The root broadcasts sync beacons in the inactive period for new (unsynchronized) nodes to join the superframe architecture.

Initial Synchronization

While the method above is effective once the entire network is synchronized, there are a couple more considerations to make when the network is not fully synchronized or when a new node joins the system. Failing to do so will cause a node to continuously miss the beacons that advertise transmission schedules.

To this end, as we show in Figure 5.3, an unsynchronized node wakes up periodically and monitors the wireless medium, similar to LPL operations. In the mean time, during the idle times in the aforementioned superframe while the low-power nodes are sleeping, the root broadcasts *sync beacons* continuously, which contains only the start time of the next *regular beacon*. Once an unsynchronized or newly joined node hears this message, it also enters radio sleep mode until the next beacon interval. This scheme allows nodes to maintain

their low-power sleep cycles while quickly joining the synchronized network. To achieve reliable *sync beacon* delivery, we set the initial sleep interval of unsynchronized nodes to be much smaller than the superframe interval. This allows new nodes to put multiple efforts for picking up the *sync beacon* message, resulting in robust synchronization under dynamic link conditions.

Lastly, the high-power root does not transmit a *sync beacon* for every inactive period but only at a larger periodical intervals (e.g., 100 superframe intervals). This prevents the procedure from monopolizing the wireless medium. Since all nodes in each network use a single superframe and the *sync-beacon* occupies the channel only shortly, *MarketNet2.0* can construct a larger network consisting of multiple roots.

5.2.2 IPv6 and Routing Layers in MarketNet2.0

Neighbor Forwarding Suppression

From the preliminary evaluations of *MarketNet1.0*, we noticed that even when a destination node successfully receives a downward packet from the root, the ACK delivery towards its neighbors could fail due to natural link dynamics, such as external channel noise in our environment (Figure 3.5). This unreliable ACK delivery results in unnecessary local packet retransmissions, which lowers the nodes' energy efficiency. The unnecessary local retransmission also comes from high node densities, since each node has many neighbor nodes.

Since inefficient neighbor forwarding causes uplink period contention, *MarketNet2.0* implements a scheme where a destination's neighbor node will *probabilistically* suppress downward packet forwarding based on the expected number of neighbors of the destination node. Assuming that we aim to deliver a missing downward packet to the destination with N neighbors with a successful delivery probability denoted as P_{succ} , each neighbor node suppresses its transmissions with probability of P_{supp} , determined by

$$P_{supp} = (1 - P_{succ})^{\alpha/N} \quad (5.1)$$

where α is a predefined parameter which balances reliability and transmission overhead. With increasing α , each node aggressively participates in neighbor forwarding, which impacts the reliability positively (more retransmissions) at first, and negatively (more packet drops due to congestion) after some point. This neighbor retransmission suppression allows for best-effort downlink packet delivery with minimal traffic overhead in dynamic channel conditions. To distribute the neighbor count N to the neighboring nodes, we include this information in the routing beacon messages used for multihop routing (as an optional field in the RPL DODAG Information Object (DIO) messages).

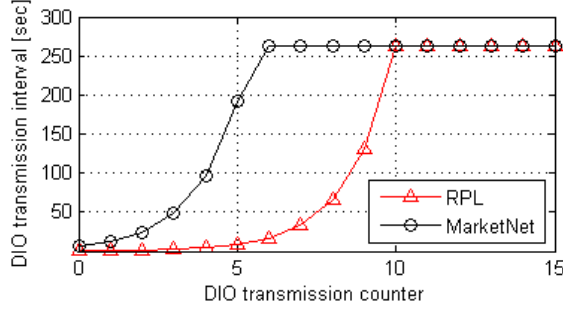


Figure 5.4: DIO transmission interval vs. DIO transmission counter for RPL and *MarketNet2.0*

DIO Transmission Interval Adjustment

RPL uses DIO messages for route advertisement to construct multi-hop routes to the root, and the trickle timer is used to control the DIO transmission interval [18, 83]. This allows RPL to achieve both low overhead and fast route recovery. For this purpose, the trickle timer initializes the DIO interval to be small, and doubles the timer after each DIO transmission until a maximum value (e.g., 256 msec and 262 sec in TinyRPL, respectively [68]) is reached.

However in *MarketNet2.0*, if the DIO transmission interval is smaller than the superframe interval, severe contention can occur during the uplink period, given that multiple DIOs can be stacked at the packet queue during the inactive period. Through our preliminary studies, we observed that *MarketNet2.0* suffers from DIO collisions during the initial phases since all nodes transmit DIOs at the minimum interval when joining the network. To overcome this issue, we configure the initial DIO transmission interval as the superframe interval in *Mar-*

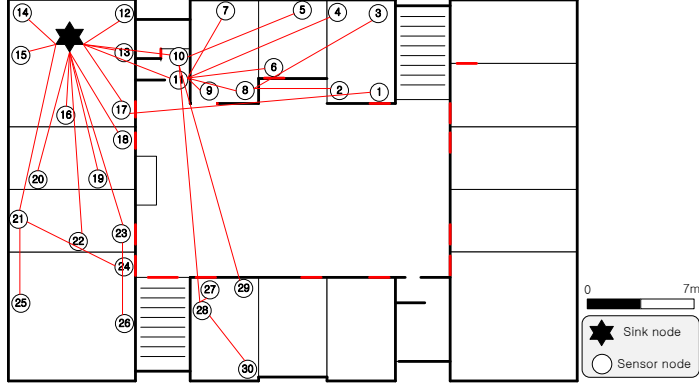


Figure 5.5: Topology map of indoor 30-node testbed with a snapshot of RPL’s routing paths.

ketNet2.0. Figure 5.4 plots the DIO transmission interval of RPL and *MarketNet2.0* as DIO transmissions continue. This cross-layer approach allows *MarketNet2.0* to quickly construct and recover its base routing topology without causing congestion from DIO transmissions.

5.3 System Evaluation

We now present empirical evaluations of *MarketNet2.0* using an indoor testbed and a market environment as in Chapter 3.

5.3.1 Testbed Evaluations

Figure 5.5 presents the topology of our testbed where a total of 31 nodes were deployed on a single floor office with one node acting as the root of the network, resulting in a 4-hop network. Using this testbed, we first present results for the packet loss rate, duty-cycle

and networking overhead for varying sleep intervals.

In this experiment, we generate periodic uplink packets (from the low-power nodes to the root) at an interval of 450 seconds, while sending downlink packets (from the root to low-power nodes) at an interval of 90 seconds (i.e., traffic rate of 3 seconds/packet at the root). We select such a balance between the two types of traffic based on an interview with market managers. Furthermore, for RPL and *MarketNet1.0*, we vary the sleep intervals of the underlying LPL operations to be 0.5, 1, 2 and 2.5 seconds so that the LPL interval is below the traffic rate of 3 seconds/packet at the root. We empirically set $\alpha = 2$ for *MarketNet2.0* to minimize transmission overhead while providing reliable downward packet delivery.

We use a superframe interval of 6 seconds for *MarketNet2.0*, and the downlink and uplink transmission durations are configured to be 90 and 120 msec, respectively, allowing for a steady 5.79 seconds of radio off time per superframe interval. We select these values for two major reasons. First, our goal was to achieve at least 3 months lifetime for our price tags on two AA batteries based on the interviews with the market managers. Given that a typical AA battery has a capacity of 900 mAh, and calculating for ~ 60 mW of active power consumption on our nodes, our target duty-cycle was 3.5% or lower. Secondly, from our literature survey which suggested a price update throughput of over 5000 messages per hour [6], our target downlink throughput is 100 packets per minute. This requires for at least a 90 msec downlink

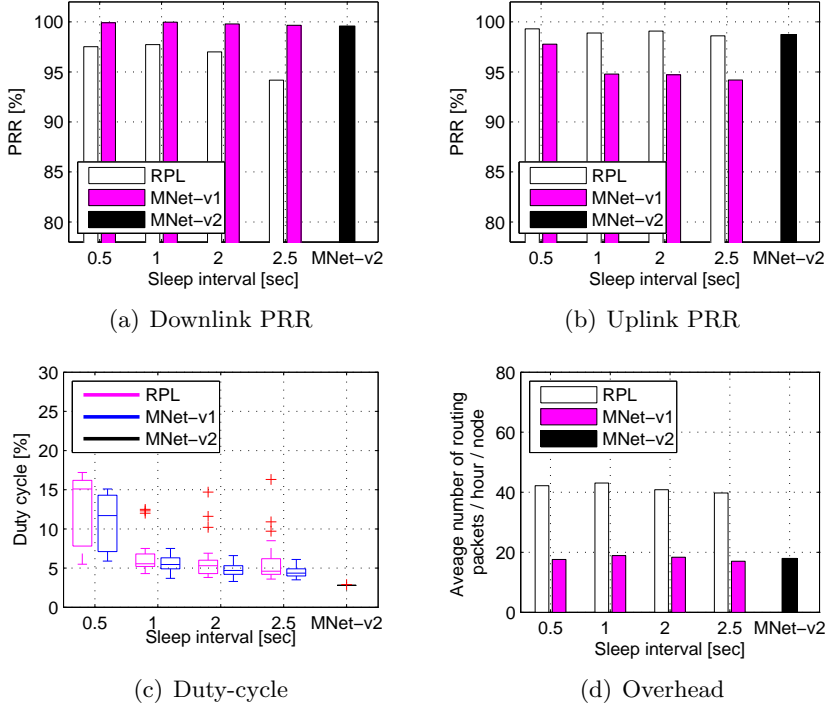


Figure 5.6: Loss-rate and radio duty-cycle results from the testbed for varying sleep intervals and *MarketNet2.0*. *MarketNet2.0* provides the lowest networking overhead and duty-cycles.

period each 6 seconds. The uplink period duration was configured to allow sufficient time for forwarding the given uplink traffic (for all low-power nodes) over multiple hops.

As a result, as Figure 5.6(c) shows, this configuration leads to achieving a radio duty-cycle of 3.5% for all low-power nodes in *MarketNet2.0*. In contrast, the radio duty-cycles of RPL and *MarketNet1.0* converge at a higher value due to the transmission inefficiency and networking overhead (Figure 5.6(d)) while maintaining the same level

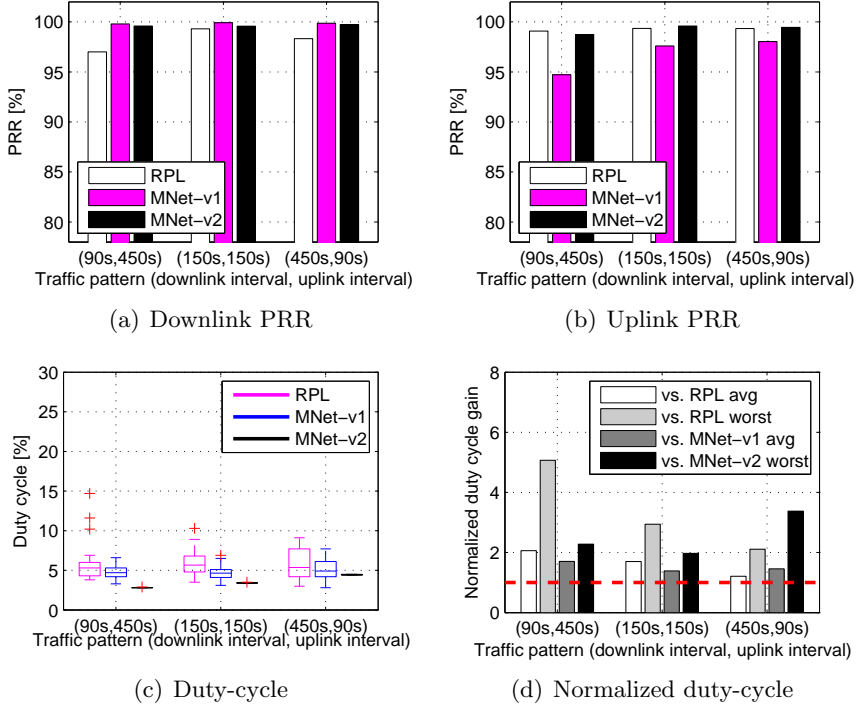


Figure 5.7: Loss-rate and duty-cycle results from the testbed for varying uplink and downlink traffic interval patterns. *MarketNet2.0* shows the lowest packet loss and duty-cycles.

of reliability as *MarketNet2.0*. If we configure the LPL sleep interval of RPL and *MarketNet1.0* higher hoping to improve the duty-cycle, nodes would not be able to handle the given traffic and the duty-cycle will further increase. We note that *MarketNet2.0*'s superframe architecture allows us to further adjust its radio duty-cycle with respect to the energy consumption of the e-price tags' sensors and display units to match the target lifetime.

We can see from Figures 5.6(a) and 5.6(b) that *MarketNet2.0*

shows reduced uplink packet loss ratios compared to the other two protocols. Especially, compared to *MarketNet1.0*, we noticed that the explicit separation of uplink and downlink packets allows the upwards packet delivery performance to match that of RPL. Given that RPL is used as its basis, this is the most ideal performance for *MarketNet2.0*.

On a different perspective, we vary the uplink and downlink traffic intervals (i.e., [90sec, 450sec], [150sec, 150sec] and [450sec, 90sec]), while maintaining an LPL sleep interval of 2 seconds. Under such conditions, Figures 5.7(a) and 5.7(b) show that the performance of *MarketNet2.0* takes the best of RPL and *MarketNet1.0* for both uplink and downlink traffic. The benefits of reduced radio duty-cycle also holds in this experiment as we plot in Figures 5.7(c) and 5.7(d). Figure 5.7(d) plots the normalized duty-cycles of RPL and *MarketNet1.0* against the duty-cycle of *MarketNet2.0*. In many cases, the duty-cycle of *MarketNet2.0* outperforms others by two-fold.

Finally, on the testbed, we examine the impact of utilizing uplink partitioning (c.f., Section 5.2.1) using Figures 5.8(a) and 5.8(b). Here, we compare the performance of *MarketNet2.0* with and without uplink partitioning. These experiments were performed during the night-time to focus solely on the effect of congestion. The results show that uplink partitioning improves both link layer ETX and uplink packet delivery performance. The improvement becomes more significant with increasing uplink traffic, which confirms that our uplink partitioning scheme reduces packet collisions by designating slots

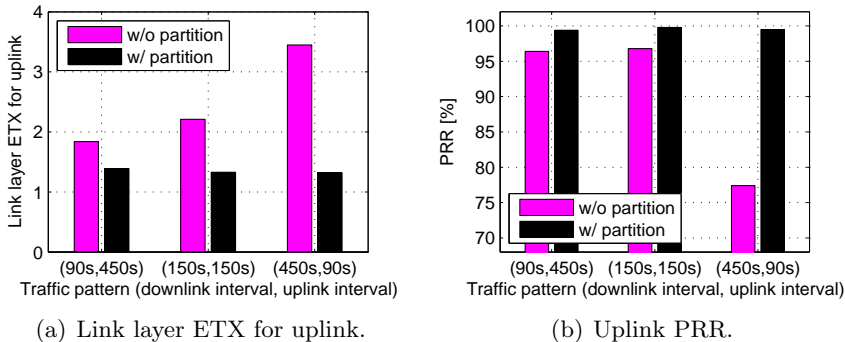


Figure 5.8: Impact of uplink partitioning. Uplink partitioning minimizes both the per-link ETX and packet loss ratios for uplink traffic by separating transmission slots.

for the nodes' transmissions.

Overall the results from the testbed suggest that *MarketNet2.0*, on an operational perspective, successfully addresses the performance limitations of RPL and *MarketNet1.0*.

5.3.2 Market Deployments

For evaluating *MarketNet2.0* in the market, we deploy nodes identically to our preliminary measurements in Section 3.3 (c.f., Figure 3.11). Furthermore, we select the same set of networking parameters as in the testbed experiments (e.g. LPL sleep interval of 2 seconds). While experiments were not performed simultaneously (i.e., 11AM-9PM on different days), which could lead to inconsistent results due to potential interference from other systems and differences in WiFi activities over different wireless channels, we confirmed that

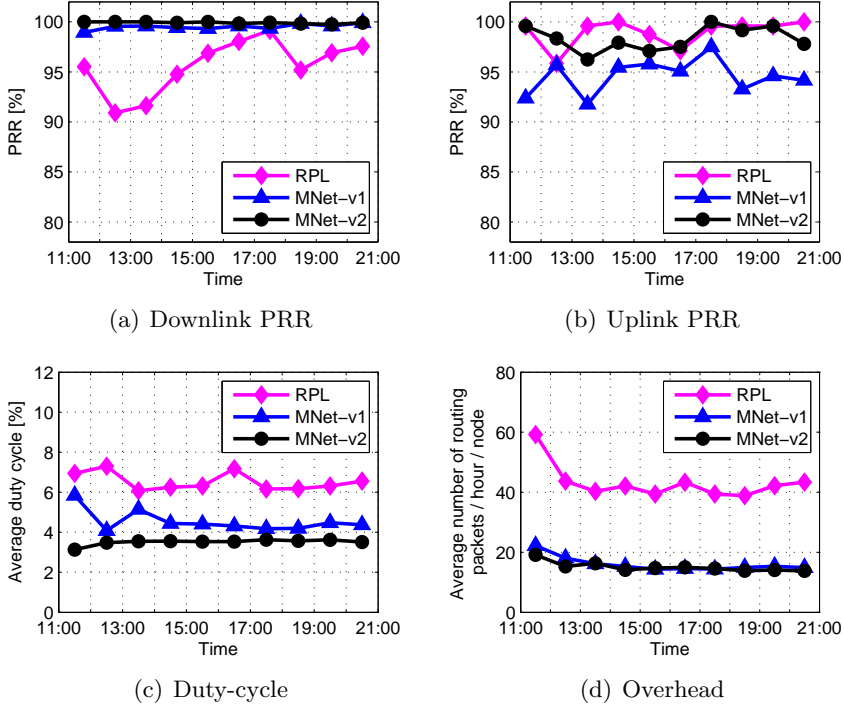


Figure 5.9: Network performance over 10-hour period for RPL, *MarketNet1.0* and *MarketNet2.0* in the market environment. While the real market environment introduces an additional level of fluctuation over time, the performance trends of *MarketNet2.0* match our testbed results.

for the three days of testing (e.g., one for each system), the number of market customers were roughly similar.

We first present the packet loss ratio for uplink and downlink traffic in Figures 5.9(a) and 5.9(b), respectively. We noticed that *MarketNet2.0* maintains reliable ($< 1\%$) downlink packet delivery regardless of the number of customers during the day, as *MarketNet1.0* does. In contrast, the downlink performance of RPL fluctuates mostly due

to dynamic channel conditions during busy times, especially due to human activities. During these periods, link qualities may change frequently, but RPL's route changes cannot keep pace with the link fluctuations. Therefore, even if a RPL node changes its route with respect to the fluctuations, without frequent DAO updates, the parent node is unaware of changes, leading to non-optimal path selection and packet losses for downward traffic. On the other hand, uplink performances of all three protocols vary as time passes due to unstable low-power links in dynamic environments as discussed in Section 3.2, where *MarketNet1.0* provides the worst performance.

Specifically, the main reason behind the slightly better performance of RPL in some cases compared to *MarketNet2.0* is owing to LPL used under RPL. A link layer transmission of LPL involves a set of repetitive transmissions for a sleep interval, which produces a dense retransmission effect. With retransmissions at the link and LPL layers, RPL holds a higher chance of delivering packets in dynamic channel conditions. Furthermore, since RPL also uses DAO messages to determine link qualities, it has a higher chance of selecting higher quality links compared to *MarketNet1.0* and *MarketNet2.0*. Nevertheless, Figure 5.9(c) shows that *MarketNet2.0* achieves the lowest duty-cycle by sacrificing such retransmissions and removing DAO overhead (as plotted in Figure 5.9(d)).

We analyze the performance of high power transmissions in *MarketNet2.0* using Figures 5.10(a) and 5.10(b). Firstly, high power

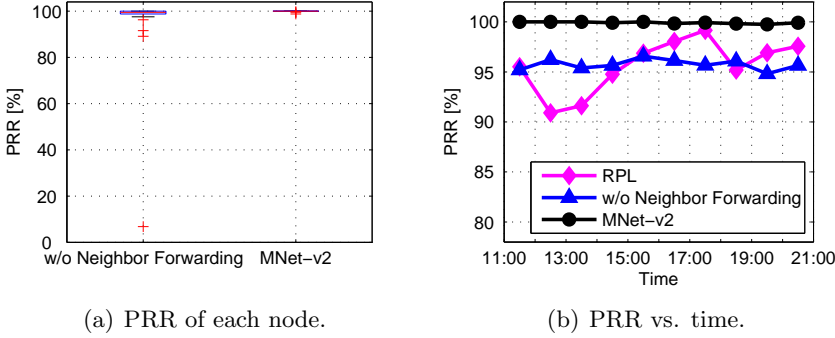


Figure 5.10: Downlink PRR of *MarketNet2.0* with and without neighbor forwarding scheme. Neighbor forwarding significantly improves reliability when transmission of high power root suffers from link dynamics or path loss.

root successfully transmits 95% of downwards packets via single hop. However, direct downward transmissions suffer from unfair reliability among nodes due to different path loss, which leads to 93.23% of loss rate for the worst node (i.e., node 25) as shown in Figure 5.10(a). Furthermore, Figure 5.10(b) shows that link dynamics impact the performance of high power transmission, which causes unstable reliability in the time domain. These observations confirm that neighbor forwarding (i.e., local retransmission) is necessary to achieve reliable downward packet delivery in *MarketNet2.0*.

We now look deep into the performance of the three protocols, especially on the packet transmission perspective at the low-power nodes using Figure 5.11(a). The number of DIO packets used to maintain the base multihop topology is almost identical, implying that the overall network stability of the systems is similar despite testing them

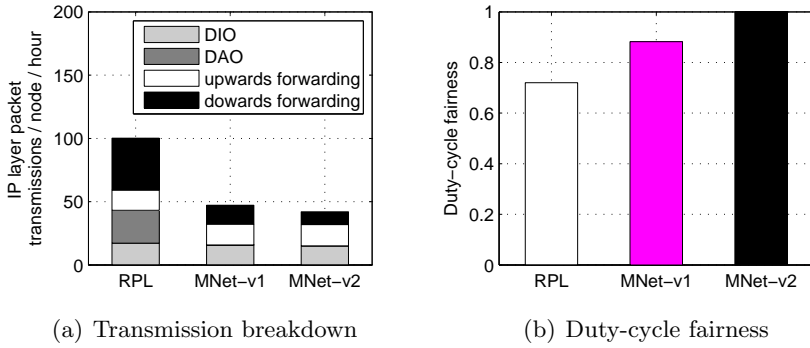


Figure 5.11: Transmission overhead and duty-cycle fairness in the market for RPL, *MarketNet1.0* and *MarketNet2.0*. The reduction in transmission overhead is due to the suppression of DAO transmissions and packet forwarding for downwards delivery. Furthermore, *MarketNet2.0* shows a fair duty-cycle performance among the deployed nodes.

on three different days. By suppressing DAO transmissions, *MarketNet1.0* and *MarketNet2.0* successfully reduce the traffic overhead on low-power nodes. Furthermore, *MarketNet1.0* and *MarketNet2.0* allows low-power nodes to forward downlink packets only when they detect failure of high-power transmissions, which noticeably reduces the frequency of downward forwarding.

Lastly, Figure 5.11(b) plots the Jain’s Fairness Index for the radio duty-cycle of low-power nodes in the network. We see that the use of superframes and synchronized wake-ups lead the nodes in *MarketNet2.0* to achieve a fair duty-cycle. On the other hand, the multihop packet forwarding participation forces RPL to show the lowest fairness in terms of duty-cycle distribution among nodes. The low and fair duty-cycle of *MarketNet2.0* synchronizes the lifetime of all low-power

nodes. It also minimizes the need for irregular human intervention, compared to RPL and *MarketNet1.0*, where some nodes deplete their batteries earlier than the others.

Comprehensively, our evaluations show that the performance of *MarketNet2.0* takes the positive ends of both RPL and *MarketNet1.0* in several ways. Compared to RPL, *MarketNet2.0* significantly improves downlink performance with much lower transmission overhead. Compared to *MarketNet1.0*, *MarketNet2.0* provides greatly improved uplink performance. Most importantly, *MarketNet2.0*'s network-wide synchronization allows nodes to enjoy a longer and fairer lifetime.

In our evaluations, we compare *MarketNet2.0* with RPL because, given that is considered to be the *de-facto* standard IPv6 routing protocol for low-power and lossy networks. As alternative comparisons, recently proposed networking protocols such as ORPL [84] or LWB [85] can also be considered. Nevertheless, ORPL in which nodes in the network “anycast” messages through the RPL DODAG, the fairness between the nodes and routing overhead cannot outperform RPL. Moreover, the anycast overhead of ORPL can lead to sacrificing the radio duty-cycle for improved packet delivery performance over RPL. As for LWB in which the Glossy protocol [86] is used as its basis, we believe it could potentially be a candidate protocol that can be compared to *MarketNet2.0* to see their respective advantages and disadvantages. We leave this as an interesting future work.

5.4 Non-technical but Practical Lessons

This work includes lots of experiments in a real-world urban marketplace. This section presents some non-technical, but practical lessons that we have learned from the real-world deployment experiences.

First, we need to be very talkative with market staffs to be friendly with them. A supermarket has many staffs each of whom covers several shelves, thus we frequently met them. However, most of them were not familiar with our wireless devices and initially thought that we were tapping. To do experiments in markets, we must persuade the staffs and get along with them.

Second, we need to be sincere customers for the targeted market. That is, we should continuously buy products and eat food in the targeted market. This behavior can show that we are not free riders and make the market staffs more kind to us.

Third, given that urban markets are extremely crowded, we must protect our precious devices from curious children customers to save time and money. Cleaners working in markets also possibly throw away the unfamiliar devices. In practice, during our experiments, we lost one sensor mote and one laptop.

Lastly, since this work is not part of an official project but a self-motivated work, we were required to find a marketplace where we would do our experiments, by ourselves. We privately contacted market managers and got approval. Thus, if we consider a real-world

experiment, we have to build up a proper ‘*social*’ network to find an experiment field before starting our main work, i.e., design of a ‘*wireless*’ network.

5.5 Summary

In this chapter, we present *MarketNet2.0* which improves *MarketNet1.0*. Given that *MarketNet1.0* suffers from uplink PRR degradation due to lack of spatial reuse, we replace LPL, asynchronous low-power link layer protocol of *MarketNet1.0*, with network-wide superframe architecture, which maintain other design elements. Our key idea is to use high power root transmission for network-wide time synchronization as well as single hop downlink packet delivery. Based on the time synchronization, we allow all nodes in the network to share a single superframe in which low-power nodes’ transmission and high power root transmission are separated in the time domain, which improves uplink PRR. We evaluate the performance of RPL, *MarketNet1.0*, and *MarketNet2.0* through both indoor testbed experiments and crowded market deployments. Our results reveal that *MarketNet2.0* can achieve the best performance among the competitive networks. Specifically, *MarketNet2.0* provides similar downlink PRR to *MarketNet1.0* (better than RPL), similar uplink PRR to RPL (better than *MarketNet1.0*), and the lowest radio duty-cycle with perfect duty-cycle fairness. Through this extensive performance eval-

uation, we verify that *MarketNet2.0* is suitable to support our target application, wireless and remote reconfiguration of e-price tags, by overcoming environmental challenges in urban crowded markets and fulfilling major application requirements.

Chapter 6

Conclusion

6.1 Research Contributions

For over a decade, many wireless systems have been designed to automate routine tasks that were previously performed manually. Our work builds on past works by showing how low-power wireless systems can help improve a busy market environment using electronic, wirelessly reconfigurable price tags. This work started with a critical examination of existing networking architectures that could potentially be used for such applications. We performed experiments to gather empirical data and gauge how real-world wireless environments impact their performance. Furthermore, we present *MarketNet*, which addresses the challenges of the application and environments, then evaluate *MarketNet* through a deployment in an actual market. We envision that by addressing additional systematic and deployment-

specific challenges, and combining such experiences with the *MarketNet* network architecture, we will be able to enhance *MarketNet*'s practical utility in busy market environments.

On a network perspective, the key points of our work are in proposing the use of asymmetric (heterogeneous) transmission power to achieve both (1) downlink reliability and (2) low-power end devices for battery operation. In other words, if we increase the transmission power of the end devices to form a single-hop topology, this would sacrifice the nodes' energy efficiency. On the other hand, reducing the root's transmission power and forming a symmetric multi-hop topology, would reduce the downlink reliability as in the RPL experiments. Our goal is to achieve the best of both worlds, given the constraints introduced from our devices and the application itself.

Although *MarketNet*'s target application is e-price tagging system, we see that our work on asymmetric transmission power-based networks 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. Below are several applications which can take advantage of the *MarketNet* architecture.

- **Wireless Reprogramming:**
- **Emergency Message Broadcasting:** Various applications are

designed to alert inhabitants in a target environment where emergency situations arise [87]. To achieve rapid delivery of emergency messages, flooding can be employed which requires all nodes participate in forwarding [81, 16]. The resultant multi-hop latency, coupled with duty cycling of nodes to conserve energy can degrade overall system performance [88, 89]. APN provides an efficient alternative for propagating alert messages.

- **Mobile Applications:** With emerging cyber physical systems (CPS) 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 bi-directional route establishment challenging [90]. In APNs, mobile nodes need to keep their routing entries only towards the gateway, since downward transmissions are completed using a high-power single-hop link.

6.2 Further Research Direction

The purpose of this work was to propose a suitable networking solution for e-price tagging applications given their system-level requirements. We emphasize that we have no intention to claim that our solution is complete, but rather a step towards building a better solution. We

summarize some interesting future research directions below.

- **Scalability:** There is always a possibility that a single root and its associated APN cannot cover the entire market environment as a single network. In such cases, the use of multiple roots, interconnected in a tiered architecture would be more suitable [11]. In such cases, each root can use a orthogonal superframe with other root nodes either in the time or frequency domain to avoid interference. *MarketNet* allows for this extension to take place easily, but the exploration of practical and systematic issues regarding such deployments is left as future work.
- **Node Density:** Although we test the performance of *MarketNet* using 30 nodes, in practice, price tags are more densely deployed in urban markets. (e.g., +10k tags in our test field). *MarketNet* addresses this challenge using its uplink period partitioning and neighbor forwarding suppression. Moreover, a wireless module possibly covers multiple e-price tags in a row or even an entire rack by connecting them with wired communication. While we explore the initial steps to form a networked system for the market environment, at-scale testing and how we can resolve any systematic issues in the deployments remains as an important next step of research.
- **Uplink Performance Enhancement:** Although this work aims to provide reliable downlink performance, improvement of uplink performance is also valuable future work. To this end, we may ex-

exploit the high power root transmission and/or more advanced routing metrics such as cETX (which considers spatiotemporal correlation) [91] to find better upward routes with lower control overhead. Furthermore, given that *MarketNet* synchronizes all nodes in the time domain, design of resource (time slot) scheduling mechanism for uplink transmissions can improve reliability with further energy saving [92].

- **Transport Protocol:** While *MarketNet* achieves 99.9% and 98.3% data delivery reliability for downlink and uplink traffic (98.7% and 93.8% for the worst nodes), respectively, for some markets, this is still the *best effort* approach and may not be enough. For guaranteed reliable transport, standard TCP or other reliable protocols such as RCRT [93] can be employed on top of *MarketNet* to complete the price updating procedures.
- **Node Placement:** Optimally provisioning an e-price tag network given a store layout by performing offline analysis and then placing the devices would be an interesting direction for future research as well. While we do not explore this direction in the scope of this work, we conjecture that this will be a challenging approach given the dynamics of the wireless environment within the market.
- **E-price Tag Implementation:** Although we focus on providing a proper network architecture to support e-price tag system, implementation of e-price tags is still an interesting and important

future work. To this end, we need to combine a low power wireless module with a low power display module. Furthermore, we need to consider how to make each e-price tag sense the rack status where it is deployed. Using weight or infrared ray sensors may be one of possible approaches.

Bibliography

- [1] J. Ko, J. Lim, Y. Chen, R. Musaloiu-E., A. Terzis, G. Masson, T. Gao, W. Destler, L. Selavo, and R. Dutton, “MEDiSN: Medical Emergency Detection in Sensor Networks,” *ACM Transactions on Embedded Computing Systems*, 2010.
- [2] J. Paek, J. Hicks, S. Coe, and R. Govindan, “Image-Based Environmental Monitoring Sensor Application Using an Embedded Wireless Sensor Network,” *Sensors*, vol. 14, no. 9, pp. 15 981–16 002, 2014.
- [3] R. Szewczyk, A. Mainwaring, J. Anderson, and D. Culler, “An Analysis of a Large Scale Habitat Monitoring Application,” in *Proc. ACM SenSys’04*, Nov. 2004.
- [4] G. Tolle, J. Polastre, R. Szewczyk, N. Turner, K. Tu, P. Buonadonna, S. Burgess, D. Gay, W. Hong, T. Dawson, and D. Culler, “A Macroscopic in the Redwoods,” in *Proc. ACM SenSys’05*, Nov. 2005.

- [5] N. Xu, S. Rangwala, K. K. Chintalapudi, D. Ganesan, A. Broad, R. Govindan, and D. Estrin, “A Wireless Sensor Network for Structural Monitoring,” in *Proc. ACM SenSys’04*, Nov. 2004.
- [6] J. G. Evans, R. A. Shober, S. A. Wilkus, and G. A. Wright, “A Low-cost Radio for an Electronic Price Label System,” *Bell Labs Technical Journal*, vol. 1, no. 2, pp. 203–215, 1996.
- [7] K. Yu, Z. Xie, J. Qian, and G. Jin, “The Implementation of Electronic Intelligent Tag System based on Wireless Sensor Network,” *Communications and Network*, vol. 5, pp. 39 – 43, 2013.
- [8] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, “Next Century Challenges: Scalable Coordination in Sensor Networks,” in *Proc. MobiCom’99*, pp. 263–270, Aug. 1999.
- [9] D. Culler, P. Dutta, C. T. Ee, R. Fonseca, J. Hui, P. Levis, J. Polastre, S. Shenker, I. Stoica, G. Tolle, and J. Zhao, “Towards a Sensor Network Architecture: Lowering the waistline,” in *Proc. HotOS’05*, pp. 139–144, Jun. 2005.
- [10] J. Polastre, J. Hui, P. Levis, J. Zhao, D. Culler, S. Shenker, and I. Stoica, “A Unifying Link Abstraction for Wireless Sensor Networks,” in *Proc. ACM SenSys’05*, pp. 76–89, Nov. 2005.
- [11] J. Paek, B. Greenstein, O. Gnawali, K.-Y. Jang, A. Joki, M. Vieira, J. Hicks, D. Estrin, R. Govindan, and E. Kohler, “The

- Tenet Architecture for Tiered Sensor Networks,” *ACM Transactions on Sensor Networks*, vol. 6, no. 4, pp. 34:1–34:44, Jul. 2010.
- [12] T. He, J. Stankovic, R. Stoleru, Y. Gu, and Y. Wu, “Essentia: Architecting Wireless Sensor Networks Asymmetrically,” in *Proc. IEEE INFOCOM’08*, Apr. 2008.
- [13] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, “Energy-efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet,” in *Proc. ACM ASPLOS’02*, pp. 96–107, 2002.
- [14] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, “Wireless Sensor Networks for Habitat Monitoring,” in *Proc. ACM WSNA’02*, pp. 88–97, 2002.
- [15] G. Tolle, J. Polastre, R. Szewczyk, D. Culler, N. Turner, K. Tu, S. Burgess, T. Dawson, P. Buonadonna, D. Gay, and W. Hong, “A Macroscopic in the Redwoods,” in *Proc. ACM SenSys’05*, pp. 51–63, 2005.
- [16] G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh, “Fidelity and Yield in a Volcano Monitoring Sensor Network,” in *Proc. OSDI’06*, Nov. 2006.
- [17] S. Kim, S. Pakzad, D. Culler, J. Demmel, G. Fenves, S. Glaser, and M. Turon, “Health Monitoring of Civil Infrastructures Using

- Wireless Sensor Networks,” in *Proc. IPSN’07*, pp. 254–263, Apr. 2007.
- [18] T. W. Ed., P. T. Ed., A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik, J. Vasseur, and R. Alexander, “RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks,” *RFC 6550*, Mar. 2012.
- [19] G. Montenegro, N. Kushalnagar, J. Hui, and D. Culler, “Transmission of IPv6 Packets over IEEE 802.15.4 Networks,” *RFC 4944*, 2007.
- [20] J. Ko, S. Dawson-Haggerty, O. Gnawali, D. Culler, and A. Terzis, “Evaluating the Performance of RPL and 6LoWPAN in TinyOS,” in *Proc. IPSN’11 Workshop*, Apr. 2011.
- [21] O. Gnawali, R. Fonseca, K. Jamieson, D. Moss, and P. Levis, “Collection Tree Protocol,” in *Proc. ACM SenSys’09*, pp. 1–14, Nov. 2009.
- [22] J. Ko, J. Eriksson, N. Tsiftes, S. Dawson-Haggerty, J.-P. Vasseur, M. Durvy, A. Terzis, A. Dunkels, and D. Culler, “Beyond Interoperability: Pushing the Performance of Sensor Network IP Stacks,” in *Proc. ACM SenSys’11*, pp. 1–11, Nov. 2011.
- [23] H.-S. Kim, H. Im, M.-S. Lee, J. Paek, and S. Bahk, “A Measurement Study of TCP over RPL in Low-power and Lossy Net-

- works,” *To appear in Journal of Communications and Networks*, 2015.
- [24] Cisco, “Connected Grid Networks for Smart Grid - Field Area Network,” http://www.cisco.com/web/strategy/energy/field_area_network.html, 2015.
- [25] H. Kermajani and C. Gomez, “On the Network Convergence Process in RPL over IEEE 802.15.4 Multihop Networks: Improvement and Trade-offs,” *Sensors*, vol. 14, no. 7, pp. 11 993–12 022, 2014.
- [26] U. Herberg and T. Clausen, “A Comparative Performance Study of the Routing Protocols LOAD and RPL with Bi-directional Traffic in Low-power and Lossy Networks,” in *ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks*, 2011.
- [27] T. Clausen, U. Herberg, and M. Philipp, “A Critical Evaluation of the IPv6 Routing Protocol for Low Power and Lossy Networks (RPL),” in *Proc. IEEE WiMob’11*, Oct. 2011.
- [28] D. Popa, M. Gillmore, L. Toutain, J. Hui, R. Ruben, and K. Monden, “Applicability Statement for the Routing Protocol for Low Power and Lossy Networks (RPL) in AMI Networks,” *draft-ietf-roll-applicability-ami-10*, Jan. 2015.

- [29] E. Ancillotti, R. Bruno, and M. Conti, “The Role of the RPL Routing Protocol for Smart Grid Communications,” *IEEE Communications Magazine*, vol. 51, no. 1, pp. 75–83, Jan. 2013.
- [30] D. Wang, Z. Tao, J. Zhang, and A. Abouzeid, “RPL Based Routing for Advanced Metering Infrastructure in Smart Grid,” in *Proc. IEEE ICC’10 Workshops*, May 2010.
- [31] N. Bressan, L. Bazzaco, N. Bui, P. Casari, L. Vangelista, and M. Zorzi, “The Deployment of a Smart Monitoring System using Wireless Sensor and Actuator Networks,” in *Proc. IEEE Smart-GridComm’10*, pp. 49–54, Oct. 2010.
- [32] V. Gungor, B. Lu, and G. Hancke, “Opportunities and Challenges of Wireless Sensor Networks in Smart Grid,” *IEEE Transactions on Industrial Electronics*, vol. 57, no. 10, pp. 3557–3564, 2010.
- [33] E. Ancillotti, R. Bruno, and M. Conti, “Reliable Data Delivery with the IETF Routing Protocol for Low-power and Lossy Networks,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 3, pp. 1864–1877, Aug. 2014.
- [34] H.-S. Kim, J. Paek, and S. Bahk, “QU-RPL: Queue Utilization based RPL for Load Balancing in Large Scale Industrial Applications,” in *Proc. IEEE SECON’15*, Jun. 2015.

- [35] “Bluetooth 4.1 features & technical descriptions,” Bluetooth SIG, Nov. 2013, [Online]. Available: <http://www.bluetooth.com/>.
- [36] T. Lee, H.-S. Kim, M.-S. Lee, and S. Bahk, “Demo: RPL over Bluetooth Low Energy,” in *Proc. ACM SenSys’15*, pp. 467–468, Nov. 2015.
- [37] J. Ko, J. Jeong, J. Park, J. A. Jun, O. Gnawali, and J. Paek, “DualMOP-RPL: Supporting Multiple Modes of Downward Routing in a Single RPL Network,” *ACM Transactions on Sensor Networks*, vol. 11, no. 2, pp. 39:1–39:20, Mar. 2015.
- [38] J. Ryu and D. Cho, “A New Routing Scheme Concerning Power-Saving in Mobile Ad-Hoc Networks,” in *Proc. IEEE ICC’00*, pp. 1719–1722, 2000.
- [39] A. Avudainayagam, W. Lou, and Y. Fang, “DEAR: A Device and Energy Aware Routing Protocol for Heterogeneous Ad Hoc Networks,” *Journal of Parallel and Distributed Computing*, vol. 63, no. 2, pp. 228–236, Feb. 2003.
- [40] W. Liu, Y. Zhang, W. Lou, and Y. Jang, “DELAR: Device/Energy/Load Aware Relaying in Heterogenous Wireless Ad Hoc Networks,” in *Proc. IEEE MILCOM’04*, pp. 1303–1309, 2004.
- [41] W. Liu, C. Zhang, G. Yao, and Y. Fang, “DELAR: A Device-Energy-Load Aware Relaying Framework for Heterogeneous Mo-

- mobile Ad Hoc Networks,” *[IEEE Journal on Selected Areas in Communications]*, vol. 29, no. 28, pp. 1572–1584, Sep. 2011
- [42] N. Li and J. Hou, “Topology Control in Heterogeneous Wireless Networks: Problems and Solutions,” in *Proc. IEEE INFOCOM’04*, vol. 1, Mar. 2004.
- [43] S. Lin, J. Zhang, G. Zhou, L. Gu, J. A. Stankovic, and T. He, “ATPC: Adaptive Transmission Power Control for Wireless Sensor Networks,” in *Proc. ACM SenSys’06*, Nov. 2006.
- [44] Y. Huang, X. Yang, S. Yang, W. Yu, and W. Fu, “A Cross-Layer Approach Handling Link Asymmetry for Wireless Mesh Access Networks,” *IEEE Transactions on Vehicular Technology*, vol. 60, no. 3, pp. 1045–1058, Mar. 2011.
- [45] P. Zhao, X. Yang, W. Yu and X. Fu, “A Loss-Virtual-Clustering-Based Routing for Power Heterogeneous MANETs,” *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 2290–2302, Jun. 2013.
- [46] X. Du, D. Wu, W. Liu, and Y. Fang, “Multiclass Routing and Medium Access Control for Heterogeneous Mobile Ad Hoc Networks,” *IEEE Transactions on Vehicular Technology*, vol. 55, no. 1, pp. 270–277, Jan. 2006.

- [47] V. Shah and S. Krishnamurthy, “Handling Asymmetry in Power Heterogeneous Ad Hoc Networks: A Cross Layer Approach,” in *Proc. IEEE ICDCS’05*, pp. 749–759, Jun. 2005.
- [48] V. Shah, E. Gelal, and S. Krishnamurthy, “Handling Asymmetry in Power Heterogeneous Ad Hoc Networks,” *Computer Networks*, vol. 51, no. 10, pp. 2594–2615, Jul. 2007.
- [49] J. Beckert and P. Aspers, *The Worth of Goods: Valuation and Pricing in the Economy*. Oxford: Oxford University Press, 2011.
- [50] J. Hagberg and H. Kjellberg, “How Much Is It? Price Representation Practices in Retail Markets,” *Marketing Theory*, vol. 15, no. 2, pp. 179–199, 2014.
- [51] V. Shankar and M. S. Yadav, “Innovations in Retailing,” *Journal of Retailing*, vol. 87, pp. S1–S2, 2011.
- [52] D. Grewal, K. L. Ailawadi, D. Gauri, K. Hall, P. Kopalle, and J. R. Robertson, “Innovations in Retail Pricing and Promotions,” *Journal of Retailing*, vol. 87, pp. S43–S52, 2011.
- [53] T. Nagle and J. Hogan, *The Strategy and Tactics of Pricing: Pearson New International Edition*. Pearson Higher Education, 2013.
- [54] R. Radner, A. Radunskaya, and A. Sundararajan, “Dynamic Pricing of Network Goods with Boundedly Rational Consumers,”

- Proceedings of the National Academy of Sciences*, vol. 111, no. 1, pp. 99–104, 2014.
- [55] D. S. Kung, L. C. Lin, C. Shayo, and H. Dyck, “Business Analytics: IT-based System with Dynamic Pricing Algorithm,” *Business Journal for Entrepreneurs*, vol. 2013, no. 2, pp. 117–127, 2013.
 - [56] Moteiv Corporation, “Tmote Sky,” Available at <http://www.moteiv.com/products/tmotesky.php>.
 - [57] Texas Instruments, “MSP430 Ultra-low-power Microcontrollers,” 2006.
 - [58] Texas Instruments, “2.4 GHz IEEE 802.15.4 / ZigBee-ready RF Transceiver,” 2006.
 - [59] R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis, “Four-Bit Wireless Link Estimation,” in *Proc. HotNets’07*, Nov. 2007.
 - [60] A. Kara and H. L. Bertoni, “Effect of People Moving near Short-range Indoor Propagation Links at 2.45 GHz,” *Journal of Communications and Networks*, vol. 8, no. 3, pp. 286–289, Sep. 2006.
 - [61] T. A. Wysocki, “Characterization of the Indoor Radio Propagation Channel at 2.4 GHz,” pp. 84–90, 2000.

- [62] X. Zheng, Z. Cao, J. Wang, Y. He, and Y. Liu, “Zisense: Towards Interference Resilient Duty Cycling in Wireless Sensor Networks,” in *Proc. ACM SenSys’14*, pp. 119–133, Nov. 2014.
- [63] H. Lee, A. Cerpa, and P. Levis, “Improving Wireless Simulation Through Noise Modeling,” in *Proc. IPSN’07*, 2007.
- [64] P. Levis, N. Patel, D. Culler, and S. Shenker, “Trickle: A Self-regulating Algorithm for Code Propagation and Maintenance in Wireless Sensor Networks,” in *Proc. NSDI’04*, Mar. 2004.
- [65] D. Moss and P. Levis, “BoX-MACs: Exploiting Physical and Link Layer Boundaries in Low-Power Networking,” Stanford Information Networks Group, Tech. Rep. SING-08-00, 2008.
- [66] O. Landsiedel, E. Ghadimi, S. Duquennoy, and M. Johansson, “Low power, Low Delay: Opportunistic Routing Meets Duty Cycling,” in *Proc. ACM IPSN ’12*, pp. 185–196, 2012.
- [67] D. Puccinelli, S. Giordano, M. Zuniga, and P. J. Marrón, “Broadcast-free Collection Protocol,” in *Proc. ACM SenSys ’12*, pp. 29–42, Nov. 2012.
- [68] J. Ko, S. Dawson-Haggerty, D. E. Culler, J. W. Hui, P. Levis, and A. Terzis, “Connecting Low-power and Lossy Networks to the Internet,” *IEEE Communications Magazine*, vol. 49, no. 4, pp. 96–101, Apr. 2011.

- [69] M. Ceriotti, M. Corra, L. D’Orazio, R. Doriguzzi, D. Facchin, S. Guna, G. Jesi, R. Lo Cigno, L. Mottola, A. Murphy, M. Pescalli, G. Picco, D. Pregnotato, and C. Torghele, “Is there Light at the Ends of the Tunnel? Wireless Sensor Networks for Adaptive Lighting in Road Tunnels,” in *Proc. IPSN’11*, pp. 187–198, Apr. 2011.
- [70] H.-S. Kim, Y.-J. Choi, and S. Bahk, “Elimination of Multi-hop Transmission from Downlink in Low Power and Lossy Networks,” in *Proc. IEEE ICC’14*, pp. 305–310, Jun. 2014.
- [71] Y. Chen and A. Terzis, “On the Implications of the Log-normal Path Loss Model: An Efficient Method to Deploy and Move Sensor Motes,” in *Proc. ACM SenSys’11*, pp. 26–39, Nov. 2011.
- [72] K. Srinivasan, P. Dutta, A. Tavakoli, and P. Levis, “An Empirical Study of Low-power Wireless,” *ACM Transactions on Sensor Networks*, vol. 6, no. 2, pp. 16:1–16:49, Mar. 2010.
- [73] “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.
- [74] Federal Communications Commission, “FCC Power Regulation,” Available at: <http://transition.fcc.gov/Bureaus/>

EngineeringTechnology/Documents/bulletins/oet63/
oet63rev.pdf, 1993.

- [75] “Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Enhancements for Higher Throughput,” Oct. 2009.
- [76] W. Pak, K.-T. Cho, J. Lee, and S. Bahk, “W-MAC: Supporting Ultra Low Duty Cycle in Wireless Sensor Networks,” in *Proc. IEEE GLOBECOM’08*, pp. 1–5, Nov. 2008.
- [77] Y.-J. Choi, S. Park, and S. Bahk, “Multichannel Random Access in OFDMA Wireless Networks,” *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 3, pp. 603–613, Mar. 2006.
- [78] H.-S. Kim, “SHDP Simulator,” Available at:<http://netlab.snu.ac.kr/~hskim>, 2013.
- [79] MAXFOR Technology, “WSN Communication Module, MTM-CM3300MSP,” Available at: http://www.maxfor.co.kr/datasheet/MAXFOR_Digital_Brochure.pdf, 2014.
- [80] A. Woo, T. Tong, and D. Culler, “Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks,” in *Proc. ACM SenSys’03*, pp. 14–27, Nov. 2003.
- [81] M. Marot, B. Kusy, G. Simon, and A. Ledeczi, “The Flooding Time Synchronization Protocol,” in *Proc. ACM SenSys’04*, pp. 39–49, Nov. 2004.

- [82] T.-H. Kim, J.-S. Han, H.-S. Kim, and Y.-H. Lee, “Alleviation of Contention Collision in IEEE 802.15.4 Networks,” in *Proc. IEEE WCNC’13*, pp. 65–70, Apr. 2013.
- [83] P. Levis, T. H. Clausen, J. Hui, O. Gnawali, and J. Ko, “The Trickle Algorithm,” *RFC 6206*, Mar. 2011.
- [84] S. Duquennoy, O. Landsiedel, and T. Voigt, “Let the Tree Bloom: Scalable Opportunistic Routing with ORPL,” in *Proc. ACM SenSys’13*, Nov. 2013.
- [85] F. Ferrari, M. Zimmerling, L. Mottola, and L. Thiele, “Low-Power Wireless Bus,” in *Proc. ACM SenSys’12*, Nov. 2012.
- [86] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh, “Efficient Network Flooding and Time Synchronization with Glossy,” in *Proc. IPSN’11*, 2011.
- [87] K. Lorincz, D. Malan, T. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, and S. Moulton, “Sensor Networks for Emergency Response: Challenges and Opportunities,” *IEEE Pervasive Computing*, vol. 3, no. 4, pp. 16–23, Oct. 2004.
- [88] F. Wang and J. Liu, “Duty-cycle-aware Broadcast in Wireless Sensor Networks,” in *Proc. IEEE INFOCOM’09*, pp. 468–476, Apr. 2009.

- [89] K. Yildirim and A. Kantarci, “Time Synchronization based on Slow-flooding in Wireless Sensor Networks,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 1, pp. 244–253, Jan. 2014.
- [90] J. Fink, A. Ribeiro, and V. Kumar, “Robust Control for Mobility and Wireless Communication in Cyber Physical Systems with Application to Robot Teams,” *Proceedings of the IEEE*, vol. 100, no. 1, pp. 164–178, Jan. 2012.
- [91] S. M. Kim, S. Wang, and T. He, “cETX: Incorporating Spatiotemporal Correlation for Better Wireless Networking,” in *Proc. ACM. SenSys ’15*, pp. 323–336, Nov. 2015.
- [92] S. Duquennoy, B. Al Nahas, O. Landsiedel, and T. Watteyne, “Orchestra: Robust Mesh Networks through Autonomously Scheduled TSCH,” in *Proc. ACM SenSys’15*, pp. 337–350, Nov. 2015.
- [93] J. Paek and R. Govindan, “RCRT: Rate-controlled Reliable Transport Protocol for Wireless Sensor Networks,” *ACM Transactions on Sensor Networks*, vol. 7, no. 3, pp. 20:1–20:45, Oct. 2010.

초 록

대규모 마켓의 가격표 업데이트는 단순하고 반복적인 작업이지만, 여전히 대부분의 마켓에서 수동으로 이루어지고 있다. 수동 가격표 업데이트는 잦은 직원들의 실수로 인한 소비자들의 불만과 카운터의 계산 오류를 야기하므로, 상기 업데이트 과정을 자동화하면 많은 이익을 창출할 수 있다. 한편, 최근 전자 잉크와 같은 저전력 디스플레이 기술이 발전하면서 전자 무선 가격표 업데이트 서비스의 실현 가능성이 높아졌다. 본 논문에서 우리는 상기 전자 무선 가격표 업데이트 서비스를 위한 네트워크 아키텍처를 제안하고 테스트한다. 우리는 먼저 실제 마켓에서의 예비 실험들을 통해, 현존하는 네트워크 프로토콜들이 바쁜 마켓 환경에서 발생하는 독특한 문제들을 극복하지 못하고, 낮은 성능을 보인다는 것을 밝힌다. 우리는 상기 실험들을 통해 기술적인 도전과제들을 확인하고, 장애물과 인구가 많은 환경에서 하향링크 위주의 트래픽을 전송하기에 적합한 비대칭 전송 전력 기반 시스템인 MarketNet을 제안하여 상기 도전과제들을 해결한다. 우리는 하루에 5000명 이상의 손님이 방문하는 실내 마켓 환경에서 MarketNet의 성능을 평가한다. 상기 성능 평가의 결과들은 MarketNet이 타깃 환경(장애물이 많고 붐비는 마켓)에서 타깃 응용 분야(전자 무선 가격표 업데이트)를 적절히 지원할 수 있을 뿐만 아니라, RPL과 LPL과 같은 기존 프로토콜들보다 현저히 높은 패킷 전송률과 낮은 듀티사이클을 제공한다는 것을 보여준다.

주요어 : 가격표, 사물인터넷, 라우팅 프로토콜, 무선 센서 네트워크, 저전력 손실 네트워크, IPv6

학 번 : 2011-30227