

Resource Allocation for the Internet of Everything: From Energy Harvesting Tags to Cellular Networks

Robert Margolies

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2015

©2015

Robert Margolies

All Rights Reserved

ABSTRACT

Resource Allocation for the Internet of Everything: From Energy Harvesting Tags to Cellular Networks

Robert Margolies

In the near future, *objects equipped with heterogeneous devices such as sensors, actuators, and tags, will be able to interact with each other and cooperate to achieve common goals*. These networks are termed the *Internet of Things* (IoT) and have applications in healthcare, smart buildings, assisted living, manufacturing, supply chain management, and intelligent transportation. The IoT vision is enabled by *ubiquitous wireless communications* and there are numerous resource allocation challenges to *efficiently* connect each device to the network. In this thesis, we study *wireless resource allocation* problems that arise in the IoT, namely in the areas of the energy harvesting tags, termed the Internet of Tags (IoTags), and in cellular networks (mobile and cognitive).

First, we present our experience designing and developing Energy Harvesting Active Networked Tags (EnHANTs). The prototypes harvest indoor light energy using custom organic solar cells, communicate and form multihop networks using ultra-low-power Ultra-Wideband Impulse Radio (UWB-IR) transceivers, and dynamically adapt their communications and networking patterns to the energy harvesting and battery states. Using our custom designed small scale testbed, we evaluate energy-adaptive networking algorithms spanning the protocol stack (link, network, and flow control). Throughout the evaluation of experiments, we highlight numerous phenomena which are typically difficult to capture in simulations and nearly impossible to model in analytical work. We believe that these lessons would be useful for the designers of many different types of energy harvesters and energy harvesting adaptive networks.

Based on the lessons learned from EnHANTs, we present Power Aware Neighbor Discovery Asynchronously (Panda), a Neighbor Discovery (ND) protocol optimized for networks of energy harvesting nodes. To enable object tracking and monitoring applications for IoTags, Panda

is designed to *efficiently identify* nodes which are within wireless communication range of one another. By accounting for numerous hardware constraints which are typically ignored (i.e., energy costs for transmission/reception, and transceiver state switching times/costs), we formulate a *power budget* to guarantee *perpetual ND*. Finally, via testbed evaluation utilizing Commercial Off-The-Shelf (COTS) energy harvesting nodes, we demonstrate experimentally that Panda outperforms existing protocols by a factor of 2-3x.

We then consider Proportional Fair (PF) cellular scheduling algorithms for mobile users, These users experience *slow-fading* wireless channels while traversing roads, train tracks, bus routes, etc. We leverage the predictable mobility on these routes and *present the Predictive Finite-horizon PF Scheduling ((PF)²S) Framework*. We collect *extensive channel measurement results* from a 3G network and characterize mobility-induced channel state trends. We show that a user's channel state is highly reproducible and leverage that to develop a *data rate prediction* mechanism. Our *trace-based simulations* of the (PF)²S Framework indicate that the framework can increase the throughput by 15%–55% compared to traditional PF schedulers, while improving fairness.

Finally, we study fragmentation within a probability model of combinatorial structures. Our model does not refer to any particular application. Yet, it is applicable to dynamic spectrum access networks which can be used as the wireless access technology for numerous IoT applications. In dynamic spectrum access networks, users share the wireless resource and compete to transmit and receive data, and accordingly have specific bandwidth and residence-time requirements. We prove that the spectrum tends towards states of *complete fragmentation*. That is, for every request for $j > 1$ sub-channels, nearly all size- j requests are allocated j mutually disjoint sub-channels. In a suite of four theorems, we show how this result specializes for certain classes of request-size distributions. We also show that the delays in reaching the inefficient states of complete fragmentation can be surprisingly long. The results of this chapter provide insights into the fragmentation process and, in turn, into those circumstances where defragmentation is worth the cost it incurs.

Table of Contents

List of Figures	v
List of Tables	xiii
1 Introduction	1
1.1 Background	1
1.2 Contributions	6
2 Energy Harvesting Active Networked Tags (EnHANTs): Prototyping and Experimentation	13
2.1 Introduction	14
2.2 Related Work	18
2.3 Prototype and Testbed	19
2.4 Notation and Experimental Settings	22
2.5 Energy Harvesting	23
2.6 UWB-IR Communications, Medium Access Control, and Link Layer	29
2.7 Energy Harvesting Adaptive Flow Control and Tree Selection	36
2.8 Lessons Learned	42
2.9 Ongoing and Future Work	47
2.10 Conclusions	51
3 Panda: Neighbor Discovery on a Power Harvesting Budget	53
3.1 Introduction	54
3.2 Related Work	56

3.3	System Model	57
3.4	Panda Protocol	60
3.5	Optimization of Panda	65
3.6	Panda-Dynamic (Panda-D)	68
3.7	Experimental Performance Evaluation	71
3.8	Conclusions and Future Work	82
3.9	Appendices	83
4	Exploiting Mobility in Proportional Fair Cellular Scheduling: Measurements and Algorithms	91
4.1	Introduction	92
4.2	Related Work	94
4.3	Model and Problem Formulation	95
4.4	Slow-fading measurements	100
4.5	Feasible Data Rate Prediction ($\hat{\mathbf{R}}$)	105
4.6	Allocation Estimation ($\hat{\alpha}$)	110
4.7	Performance Evaluation	111
4.8	Conclusions and Future Work	119
4.9	Appendices	120
5	Performance Evaluation of Fragmented Structures: A Theoretical Study	123
5.1	Introduction	124
5.2	The Model	127
5.3	Background and Related Work	130
5.4	Asymptotic Theory of Fragmentation	132
5.5	Convergence	141
5.6	Conclusions and Future Work	146
5.7	Appendices	147
6	Conclusions	151
	Bibliography	153

Appendix: Project-based Learning within a Large-Scale Interdisciplinary Re-

search Effort	170
7.1 Introduction	172
7.2 Context and Intended Outcomes	174
7.3 Project-based Learning	175
7.4 Project Organization and Lessons Learned	179
7.5 Experiences and Feedback	185
7.6 Outcomes	190
7.7 Conclusions	192

This page intentionally left blank.

List of Figures

1.1	Internet and cellular infrastructure as the enabler of IoT applications in body area networks, transport and asset tracking, smart environments (homes, buildings, transportation, infrastructure), and advanced systems (smart grid, defense network).	2
1.2	(a) Our envisioned IoTag form factor, and (b) our IoTag mockup integrated with a flexible solar cell and a thin-film battery.	3
1.3	Mobile user trajectory along a road through 3 cellular sectors: (a) map outline and interactions with the cellular network and (b) measured values of channel quality (E_c/I_o) during 3 different drives.	4
1.4	Non-contiguous OFDM where the users share the resource via non-overlapping sub-channels (e.g, User 2's request is fragmented across sub-channels).	5
1.5	An EnHANT prototype.	6
1.6	(a) EH nodes can be attached to everyday objects (e.g., boxes) and utilize neighbor discovery protocols for inventory tracking and monitoring applications. (b) The prototype EH node stores harvested energy in a capacitor and powers the microcontroller and transceiver.	8
2.1	A software-based light control system, along with 4 EnHANT prototypes.	15
2.2	A block diagram of the EnHANT prototype, and its interactions with the testbed.	16
2.3	A schematic diagram of the EnHANTs testbed.	20
2.4	A screenshot of the EnHANT testbed monitoring system.	22
2.5	A dark box enclosure used in the light control system.	23

2.6	Solar cells integrated with EnHANT prototypes: (a) an amorphous silicon (a-Si) solar cell, and (b) a custom-fabricated organic photovoltaic (OPV).	23
2.7	Energy harvesting for a prototype with an a-Si solar cell and an organic photovoltaic (OPV): (a) generated power as a function of irradiance (light intensity), (b) time-varying irradiance based on a light energy trace from [71], (c) power harvested by an a-Si solar cell and an OPV.	24
2.8	(a) A photo of the Energy Harvesting Module (EHM), and (b) its block diagram.	25
2.9	Node energy allocation policies: (a) the day-long irradiance trace from L-3 [71] compressed to 321s and scaled by 2.1x, (b) energy harvesting rates from 4 experiment repetitions with error-bars, (c) energy spending rates under the EX, EP-1, and EP-12 policies, and (d) battery levels, experimentally measured and simulated, under the EX and EP-1 policies.	28
2.10	S-OOK modulation and parameters.	30
2.11	The UWB-IR Communication Module.	30
2.12	(a) Packet success probability as a function of packet rate for 14 and 50 byte packets with logging enabled and disabled, (b) packet success probability as a function of packet length and distance between nodes, (c) average throughput as a function of p , the transmit probability for Aloha and CSMA (100% throughput is 18.5kbps), and (d) average energy spent per successful packet transmission, as a function of p	33
2.13	Decoupled Rate Control (DRC) link policies: Using a day-long irradiance trace compressed to 700s and scaled by 20.0x from L-2, L-3, and the corresponding (a) energy harvesting rates from 10 experiment repetitions with errorbars, (b) data rates under the EX and EP-1 policies, (c) average data rates under the EP-12 policy, experimentally obtained and simulated, and (d) percentage of harvested energy used by v , under the EX and EP-12 policies.	37

2.14	FLEX flow control policies for (a) a 3-node <i>line</i> network: Using a day-long irradiance trace compressed to 720s and scaled by 60.4x from L-2, and the corresponding (b) energy harvesting rates from 8 experiment repetitions with errorbars, (c) data rates under the EX and EP-1 node energy allocation policies, and (d) average data rates, under the EX policy, for different values of r_c	38
2.15	Topology adaptation policies for (a) a 4-node ‘ <i>diamond</i> ’ network: Using a day-long irradiance trace compressed to 400s and scaled by 20.0x from L-1, L-2, and the corresponding (b) energy harvesting rates from 8 experiment repetitions with errorbars, and (c) percentage of harvested energy used by the nodes.	41
2.16	A timeline of the prototype and the testbed development stages.	42
3.1	A hardware model of the EH node (based on the TI eZ430-RF2500-SEH [5]). . .	58
3.2	Panda protocol outline: EH nodes transition between radio states (sleep, listen, and transmit) to maintain within a power budget.	61
3.3	Renewal process representing one renewal cycle for $N = 7$ nodes: all nodes begin in a memoryless sleep state and the renewal restarts after the first node to wake up completes its transmission.	63
3.4	The performance of the PCA for varying power budgets, P_b , and number of nodes, N : (a) discovery power as the percentage of the power budget ($\Phi(1)/P_b$), and (b) the approximation ratio ($U_A/\overline{U^*}$).	69
3.5	Panda experimental setup: 5 EH nodes harvest energy through the solar cells with neighbor discovery rates monitored by a listening node.	70
3.6	(a) Power consumption and transition costs for different transmission power levels for a node transitioning between the sleep, receive, and transmit states, and back to sleep. (b) Panda experimental evaluation with varying power budgets, P_b : Convergence of the experimental discovery rate (U_E) to the analytical discovery rate (U_A) for $N = 5$	71
3.7	Panda’s discovery rate with varying power budgets, P_b , and number of nodes, N	74

3.8	Panda experimental evaluation for $N = 5$: (a) CDF of per link discovery latency; comparison to SearchLight-E [24] and BD-E [115] of (b) the discovery rate and (c) the worst case latency; (d) capacitor voltage for a node with a 30mF capacitor and $P_b = 0.5\text{mW}$	77
3.9	Panda performance evaluation for $N = 5$, $P_b = 0.15\text{mW}$: Voltage level of nodes with varying capacitor sizes over 15 minutes.	78
3.10	Panda performance evaluation for $N = 5$ and $P_b = 0.5\text{mW}$: Experimental and analytical discovery rates under varying (a) transmission power (P_t) and b) discovery message length (M).	79
3.11	Panda-D experimental evaluation for $N = 3$ and homogeneous power harvesting $P_b = 0.15\text{mW}$: (a) Capacitor voltage, V_{cap} , in a clique topology. (b) Per link experimental discovery rates for a line topology after 50 hours.	81
3.12	Panda-D experimental evaluation for non-homogeneous power harvesting experiment with $N = 5$ over 24 hours: (a) Capacitor voltage, V_{cap} , and (b) resulting neighbor table.	82
3.13	Software controlled light system (originally designed in [112]) including a dark box enclosure, high-power LED driver, and an Arduino-based light controller. . .	87
3.14	Energy harvesting characterization: (a) measurement setup and (b) resulting power harvested as a function of the irradiance produced by the software controlled light system.	87
3.15	A four day experiment to find the light level where the node stays around its energy neutral point.	88
4.1	Measurement setup and the routes on which measurements were collected (see also Table. 4.2).	101
4.2	(a) Distribution of sector association times for 27 drives along routes R1–R4 and (b) measured values of the RSSI and the serving sector E_c/I_o for a drive on part of route R4 (vertical bars indicate hand-offs).	102
4.3	Comparison of E_c/I_o (measured), E_c/I_o^* (normalized), and $\widetilde{E_c/I_o^*}$ (smoothed): (a) a mobile trace from route R4 and (b) a static trace.	103

4.4	Characterization of slow-fading: (a) the CDF of the slow-fading metric (S) for mobile and static traces and (b)–(c) scatter plots for all mobile traces of the average velocity or distance traveled while connected to a sector vs. S	104
4.5	Reproducibility of measured E_c/I_o values: (a) measurements from 7 drives through a sector on route R1 (aligned in 25m segments) and (b)–(d) the correlation coefficient of E_c/I_o across all drives (with sufficient data) on routes R1, R3, and R4.	106
4.6	Illustration of the CHLS: the coverage map segments are labeled starting from 1 (at time slot 1, the user is in segment 1). At the present time slot (T_p), the user is located in one of the segments between B_{cell} and B_{max} , which fall within the coverage area of the serving sector.	108
4.7	Evaluation of the Coverage Map Prediction Mechanism (CMPM): (a) CHLS error distribution for 500 tests compared to (b) the error distribution for 500 tests of the localization scheme provided in [141]; (c) An example CMPM data rate prediction when location and velocity are determined using GPS or CHLS.	109
4.8	(a) An example of predicted data rates for a user with $\tilde{T} = 15\text{s}$ and the corresponding $\hat{\alpha}$ estimations computed by the LSE, BGE, and RRE algorithms. (b) The data rates (r_{ij}) for a test case with $K = 7$ and $\tilde{T} = 15\text{s}$	113
4.9	Statistical characterization of the maximum delay for the PF-EXP algorithm for 30 test cases with $K = 5$, $\tilde{T} = 30\text{s}$, and varying ϵ	113
4.10	(PF) ² S Framework performance for various framework instances (combinations of $\hat{\mathbf{R}}$ and $\hat{\alpha}$ prediction algorithms): statistical evaluation of 22 test cases with $K = 7$ and $\tilde{T} = 30\text{s}$, and the resulting (a) fairness gains over PF-EXP, (b) throughput gains over PF-EXP, (c) cumulative distribution function of the users' delay for the BGE algorithm, and (d) standard deviation of the users' delay.	115
4.11	(PF) ² S Framework performance gains when using RRE with CMPM-CHLS: statistical evaluation of (a) 10 test cases with $K = 10$, varying the time horizon (\tilde{T}) and (b) 20 test cases of $\tilde{T} = 30\text{s}$, varying the number of users (K).	116

4.12	(PF) ² S Framework evaluation of a test case generated using static (immobile) measurements with $\tilde{T} = 30$ s and $K = 5$: (a) the data rates r_{ij} and corresponding (b) fairness and throughput gains over PF-EXP using the BGE algorithm. . . .	117
4.13	(PF) ² S Framework performance gains when using RRE with CPM-CHLS: statistical evaluation of 10 test cases with $\tilde{T} = 30$ s, 10 mobile users, and varying number of static users.	118
4.14	(PF) ² S Framework evaluation for an example test case with $K = 5$ users, $\tilde{T} = 30$ s: (a) fairness and (b) throughput gains over PF-EXP using the BGE algorithm for varying the number of users, x , with simulated data rate prediction offsets. . . .	118
4.15	(PF) ² S Framework performance gains when using RRE with the CPM-CHLS: (a) varying the delay parameters for the (PF) ² S Framework (D_{starved}) and the PF-EXP scheduler (ϵ) for the test case given in Fig. 4.8(b), and (b) statistical evaluation of 15 test cases with $K = 10$ and $\tilde{T} = 30$ s for varying coverage map resolution values.	119
4.16	Mapping of Channel-Quality-Indicator (CQI) to feasible data rate for Galaxy S II phones, obtained from the 3GPP specifications [10].	120
5.1	An example of admission and departure processes and the resulting fragmentation.	125
5.2	Steady-state fraction of completely fragmented items as a function of M under the BFD algorithm, $K = 9$, and a parameter $q_1 = 1 - q_9$	143
5.3	Transient of the normalized average number of fragments per item for size- j requests under BFD (log scale): $K = 5$ (uniform distribution), $M = 100,000$	145
5.4	Average number of bonds in equilibrium under BFD, with $M/K = 10^5$, item sizes uniformly distributed on $\{1, \dots, K\}$	147
7.1	Student projects by semester.	177
7.2	Students involved in the EnHANTs project.	177
7.3	Students semesters.	178
7.4	Project survey results: Effects of project participation on educational objectives.	181
7.5	A Phase III conference demonstration [179] presented by M.S. and Ph.D. students.	182

7.6	A Ph.D. student mentoring an undergraduate student.	182
7.7	Undergraduate and M.S. students presenting their work to incoming M.S. students in Summer 2013.	184
7.8	A screenshot of the umbrella project website welcome page [6].	185
7.9	Project survey results: Project involvement compared with other activities. . . .	186
7.10	Project survey results: Evaluation of mentoring by Ph.D. students and faculty. .	187

This page intentionally left blank.

List of Tables

2.1	The hardware configuration for the light control system.	21
2.2	EHM Configuration	25
2.3	The packet error rates when a cell phone placed calls at varying distances from the receiver.	31
2.4	Data rates \bar{r} under different topology adaptation policies.	41
3.1	Nomenclature	59
3.2	Discovery message structure.	73
3.3	Measured prototype parameters.	74
3.4	Panda experimental parameters: (λ_A, l_A) generated using the PCA for every input (N, P_b) pair and the resulting analytical (U_A) and experimental (U_E) discovery rate.	75
3.5	Neighbor table for $N = 5$, $P_b = 0.3\text{mW}$ after 4 hours. Entry (i, j) shows the number of discoveries of node j by node i	76
3.6	Modifications to the TI eZ430-RF2500	83
3.7	Panda idle power consumption for every input (N, P_b) pair from Table 3.4: the probability of a node waking in the middle of a packet ($\Pr(Y = 2)$) and the expected portion of the power budget (P_b) consumed.	84
3.8	Panda performance evaluation: the discovery rate U_A resulting from the PCA, and the actual power consumed when ignoring the switching costs.	86
3.9	Panda-D Experimental and Predicted Parameters for the $N = 5$ experiment from Fig. 3.12.	89

3.10	Panda-D Discovery Rate Approximation for the $N = 5$ experiment from Fig. 3.12: Error rate (%) of the experimental per-link discovery rate compared to (3.20).	89
4.1	Nomenclature	97
4.2	Summary of collected measurements	101
5.1	Simulation results using the NF and BFD policies with $K = 2$, item size probabilities $q_1 = q_2 = 1/2$, and varying M : steady-state averages of the number of unfragmented size-2 items (N_{21}) compared to averages of the total number of items (N), the number of size-1 items (N_1), and the number of fragmented size-2 items (N_{22}).	133
5.2	Simulation results using the NF and BFD policies with item sizes uniformly distributed and varying K , M : steady-state probability that an item of size- K is completely fragmented, $\rho_{K,M}$.	134
5.3	Accuracy of approximate convergence times for $K = 2$.	144
7.1	The placements of 22 undergraduate and 32 M.S. students that already graduated (6 undergraduate and 5 M.S. students are yet to graduate).	188

Acknowledgments

First and foremost, I would like to thank my Ph.D. advisor, Professor Gil Zussman. I am thankful that he has pushed me to reach my intellectual limits and beyond. Throughout this journey, he has continued to support and teach me, and I would not be where I am today without him. Additionally, I would like to express my appreciation to my entire thesis committee consisting of Professors Augustin Chaintreau, Ioannis Kymissis, Debasis Mitra, and Dan Rubenstein.

I would like to extend a special thanks to Maria Gorlatova and Guy Grebla. For the first few years of my journey, Maria was a de-facto mentor to me on the EnHANTs project and showed me what it takes to work hard and succeed as a Ph.D. student. Likewise, Guy Grebla has been my most close collaborator and friend for the latter years of my Ph.D.

Although I did not get to collaborate with everyone from the Wim.net research group, I am sincerely thankful to have worked alongside Berk Birand, Tingjun Chen, Craig Gutterman, Varun Gupta, Jelena Marasevic, and Saleh Soltan. I thank them for their extraordinary support and feedback.

And of course, this journey would not have been possible without my friends. My office mates Wenjia Zhang, Atiyah Ahsan, Cathy Chen, and Lee Zhu have been amazing and I am lucky to have had them with me. There have also those who have been there for me for this journey and many before it, Joseph Luterzo, Alicia Faruolo, Frank Visagio, Austin Leech, Jeremy Cuevas, Michael Wang, and Michael Bacchioni. Also, I would like to thank the “Boston-crew” for being there for me, including Matthew Chrobak, Emily Baratta, Eric Communiello, Pat Lowe, and Rob Sobkowich. You guys have held me together through good and bad.

Finally, this thesis would not have been possible without several successful collaborations. Through my work on the EnHANTs project, I have had the unique opportunity to collaborate with and mentor many people. I am sincerely grateful to have met and worked with all of the students. Specifically, I am thankful to have worked closely with John Sarik, Baradwaj

Vigraham, Jianxun Zhu, and Gerald Stanje. I would like to thank Ashwin Sridharan, Vaneet Aggarwal, Rittwik Jana, N.K. Shankaranarayanan, and Vinay Vaishampayan for mentoring me during an internship at AT&T and the follow up work on scheduling in cellular networks. Last, but certainly not least, I am grateful to have worked with Professor Ed Coffman studying fragmentation problems.

Financial Support: The research described in this thesis was primarily supported by a National Science Foundation Graduate Research Fellowship (grant number DGE-11-44155). Additionally, it was supported in part by Vodafone Americas Foundation Wireless Innovation Project, NSF grants CCF-0964497, CNS-0916263, CNS-10-54856, NSF CIAN ERC under grant EEC-0812072, and DTRA grant HDTRA1-13-1-0021. The work in Chapter 4 began during a summer internship at AT&T Labs-Research in 2012.

*To my amazing parents Beverly and Mitchell,
and my wonderful sister Brenda,
none of this would have been possible
without your love and support.*

This page intentionally left blank.

Chapter 1

Introduction

In the near future, *objects equipped with heterogeneous devices such as sensors, actuators, and tags, will be able to interact with each other and cooperate to achieve common goals*. These networks are termed the *Internet of Things* (IoT) and have been gaining increased attention from academia and industry [20], with applications in healthcare, smart buildings, assisted living, manufacturing, supply chain management, and intelligent transportation (see Fig. 1.1). As such, it is estimated that by 2020 there will be up to 50 billion connected devices (or approximately 6.6 devices per person on earth) [56]. Therefore, the IoT is included by the U.S. National Intelligence Council in the list of six “Disruptive Civil Technologies” with potential impacts on the U.S. national interests [120].

The IoT vision is enabled by advances in computation, control, and *ubiquitous wireless communications* [99], including ultra-lower power wireless communications [44]. In this Ph.D. thesis, we focus on the *wireless resource allocation* problems that arise a few IoT domains. First, we study the Internet of Tags (IoTags). Then we consider wireless systems that can find applications in intelligent transportation systems and in opportunistic scheduling for smart cities.

1.1 Background

We start by providing background for each of the domains.

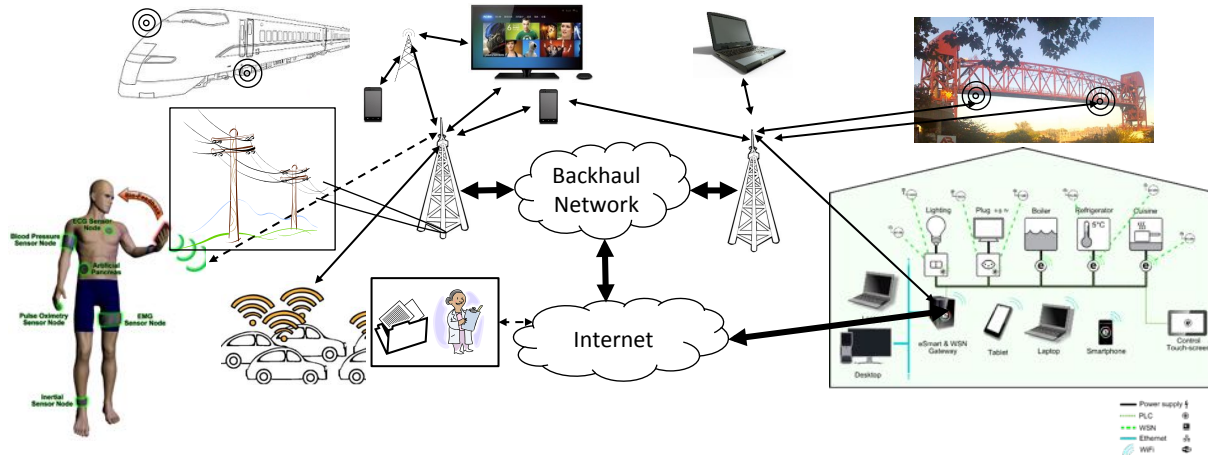


Figure 1.1: Internet and cellular infrastructure as the enabler of IoT applications in body area networks, transport and asset tracking, smart environments (homes, buildings, transportation, infrastructure), and advanced systems (smart grid, defense network).

1.1.1 The Internet of Tags

The Internet of Tags (IoTags) will be one of the enablers for the *Internet of Things* [20] and will support a variety of tracking and monitoring applications beyond what the current tracking technology, RFIDs, permits. While RFIDs make it possible to *identify* an object, IoTags will make it possible to *search* for an object, and to continuously track objects' whereabouts, including their proximity to each other.

In [60] we described a representative IoTag application: *locating a misplaced book in a library*. In this application, IoTags will be attached to library books, harvest indoor light, and wirelessly exchange IDs with the neighboring books. If a book is misplaced, its ID will be significantly different from the IDs of its neighbors, and this information will be forwarded to the librarian. Similar applications were recently discussed and demonstrated in [103,161]. They include finding specific items in a store or a warehouse, locating misplaced items, and continuous monitoring of merchandise in transit.

The realization of the IoTag vision is based on recent advances in energy harvesting [25, 123, 125, 151], which allows for dramatic reduction in the size and weight of a wireless node



Figure 1.2: (a) Our envisioned IoTag form factor, and (b) our IoTag mockup integrated with a flexible solar cell and a thin-film battery.

battery. Another enabling technology is ultra-low-power wireless communications [44, 46, 165]. The advances in these areas will allow small, flexible, lightweight tags to communicate and network in environments with limited energy availability (e.g., while harvesting indoor light energy). The form factor we envision for an IoTag is shown in Fig. 1.2(a) and a recent mock-up is shown in Fig. 1.2(b). A tag of this size (less than 4.5x4.5cm) will support continuous data rates between 1 and 13kbps [68].

However, in order to support networking such energy harvesting IoTags, hardware and algorithms on all layers of the protocol stack must be redesigned, presenting numerous challenges.

1.1.2 Mobile Cellular Scheduling for Intelligent Transportation Networks

Intelligent transportation systems are used to aide in the mobility of people (i.e., reduce congestion) while improving the safety. Recent advances in mobile computation and wireless communication systems have enabled complex systems to improve the transportation system (i.e., infrastructure-based warnings, traffic information) [122]. Additionally, in the near future, road-side cameras will be able to immediately detect and forward vehicle collision information through the cellular network to mobile users. One heavily discussed application is the use of wireless communication through the cellular network to enable automated control of a driverless transportation system, controlled through the cloud.

3G and 4G (LTE) cellular networks are an enabler of intelligent transportation systems.

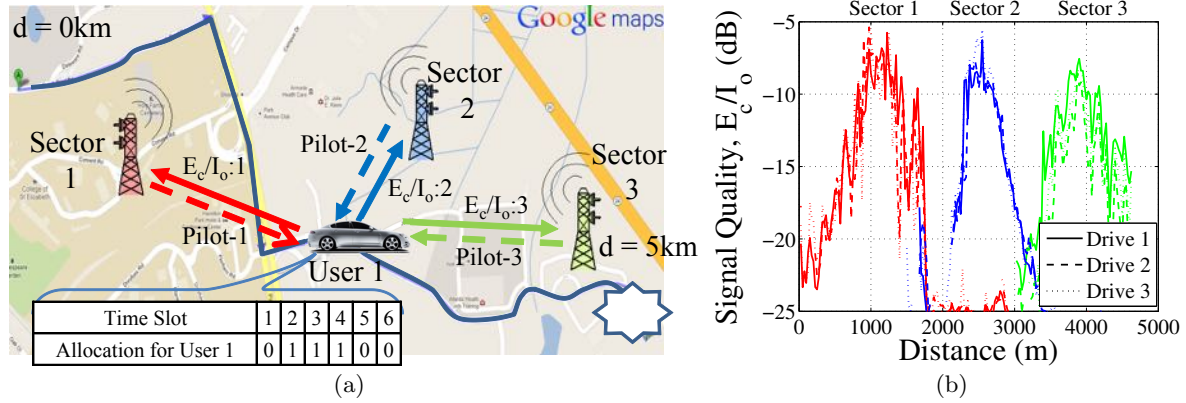


Figure 1.3: Mobile user trajectory along a road through 3 cellular sectors: (a) map outline and interactions with the cellular network and (b) measured values of channel quality (E_c/I_o) during 3 different drives.

They incorporate opportunistic schedulers [32] and allocate resources to users with good channel conditions by leveraging channel state variations, due to fast-fading,¹ as well as multi-user diversity. *Proportional Fair* (PF) scheduling algorithms are the de-facto standard for opportunistic schedulers in cellular networks [77]. They aim to provide high throughput while maintaining fairness among the users. However, existing schedulers (e.g., [18,31,91]) do not incorporate user mobility.

For example, Fig. 1.3 illustrates a trajectory of a car along a 5km path, and the signal quality (E_c/I_o) to 3 different sectors (we collected the E_c/I_o values during 3 drives on the path). As can be seen, the channel has a dominant *slow-fading* component² on which the fast-fading component is overlaid. Since PF schedulers are not optimized for mobility, we will design a scheduling framework which can account for predictable mobility.

¹Fast-fading is characterized by rapid fluctuations in the received signal strength (due mainly to multipath) [134].

²Slow-fading is characterized by slow (on the order of seconds) changes of the received signal strength (e.g., due to path loss and shadowing) [134].

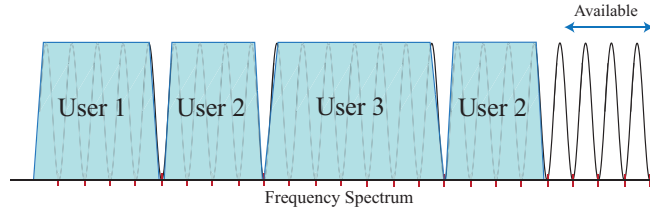


Figure 1.4: Non-contiguous OFDM where the users share the resource via non-overlapping sub-channels (e.g, User 2’s request is fragmented across sub-channels).

1.1.3 Wireless Spectrum Fragmentation

Dynamic spectrum access networks [14] (also called cognitive radio networks) have recently been proposed as a wireless access technology for smart cities [178]. In dynamic spectrum access networks, a portion of the wireless spectrum not licensed to any particular user is reserved for “opportunistic” use. To address the massive number of potential wireless devices in a smart city environment, dynamic spectrum access networks support large-scale data transmissions through opportunistic use. As described in [178], this can support applications of multimedia device communication in-home, healthcare remote monitoring, and the smart grid.

In dynamic spectrum access networks, users have to compete to transmit and receive data, and accordingly have specific bandwidth and residence-time requirements. The data transmission can take place over a channel consisting of a number of non-contiguous sub-channels. Once a transmission terminates, the corresponding sub-channel fragments are vacated and made available to other users. When no individual gap contains enough sub-channels, a user’s allocation can be broken down into fragments that fit into multiple gaps. Gaps develop randomly in both size and position as shown in Figure 1.4. Fragmentation is created solely by the random arrival and departure of user requests for varying amounts of sub-channels, and by the requirement that requests not be moved once they are allocated.

The resulting fragmentated spectrum can have a significant impact on the efficiency of real systems and networks [84, 85, 142, 174]. Therefore, we seek to provide important insights into the fragmentation process through modeling and analysis.

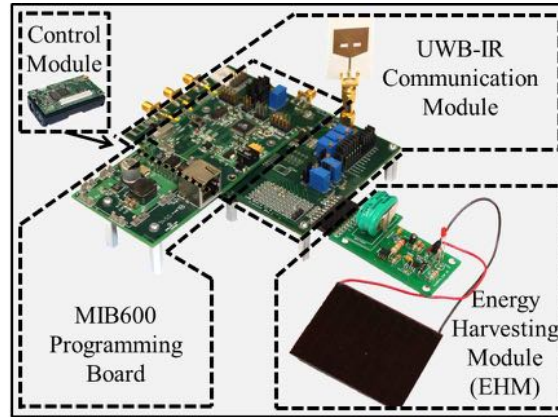


Figure 1.5: An EnHANT prototype.

1.2 Contributions

In this section, we describe the contributions made to each IoT domain.

1.2.1 The Internet of Tags

To enable the Internet of Tags (IoTags), we have taken both a top-down approach and a bottom-up approach. In the first contribution, we took a bottom-up approach to develop a network and testbed of IoTag prototypes, termed Energy Harvesting Active Networked Tags (EnHANTs). In the second contribution, we took a top-down approach to develop Panda, a neighbor discovery protocol to support the IoTags. We evaluated the performance of Panda using Commercial Off-The-Shelf (COTS) energy harvesting devices.

1.2.1.1 Energy Harvesting Active Networked Tags: Prototyping and Experimentation

In Chapter 2 we present the cross-layer design considerations of the EnHANT prototypes, which have been developed over the past 4 years through a series of design phases. We describe the current prototypes (shown in Fig. 1.5) which were designed to harvest indoor light energy using custom-designed photovoltaics, and communicate wirelessly using ultra-low-power Ultra-Wideband Impulse-Radio (UWB-IR) transceivers. We develop and evaluate networking

algorithms where the prototypes form small networks and adapt their communications patterns to the battery and energy harvesting states. The prototype design has been driven by the needs of the higher layers (e.g., flow control, multihopping), while the higher layer protocols (e.g., MAC and flow control) had to adapt to the physical layer (photovoltaic and UWB-IR) characteristics.

Moreover, we present the EnHANTs testbed, which enables controllable and repeatable experiments with communications and networking algorithms for energy harvesting nodes. The approach to testbed design and development we pioneer in this work, as well as the design of the environmental energy inputs, is applicable not just to EnHANTs, but to all future energy harvesting devices. Using the testbed and the prototypes, we conducted experiments regarding the performance of algorithms on all layers of the protocol stack, some of which were proposed in [57, 68, 89].

Insights obtained in the experiments may apply beyond the EnHANTs to energy harvesting sensor networks [86, 132, 180] and networks of computational RFIDs [73]. Throughout the evaluation of experiments, we demonstrate numerous phenomena which are typically difficult to capture in simulations and nearly impossible to model in analytical work. We summarize these lessons that were learned throughout the different phases of the prototype and testbed design process. We believe that these lessons would be useful for the designers of many different types of energy harvesters and energy harvesting adaptive networks.

The main contributions of this work are (i) the design and development of EnHANT prototypes, (ii) the design of a testbed for the evaluation of energy harvesting algorithms, and (iii) the experimental evaluation of such algorithms. To the best of our knowledge, the EnHANT prototypes are the first wireless devices that harvest energy using OPVs, the first to demonstrate multihop data forwarding over the ultra low power UWB-IR physical layer, and the first to adapt to the harvesting states in real time. We also note that this work is the first attempt to evaluate energy harvesting adaptive policies in a controllable experimental environment. We also describe these contributions in [61, 109, 112, 145, 162, 179].

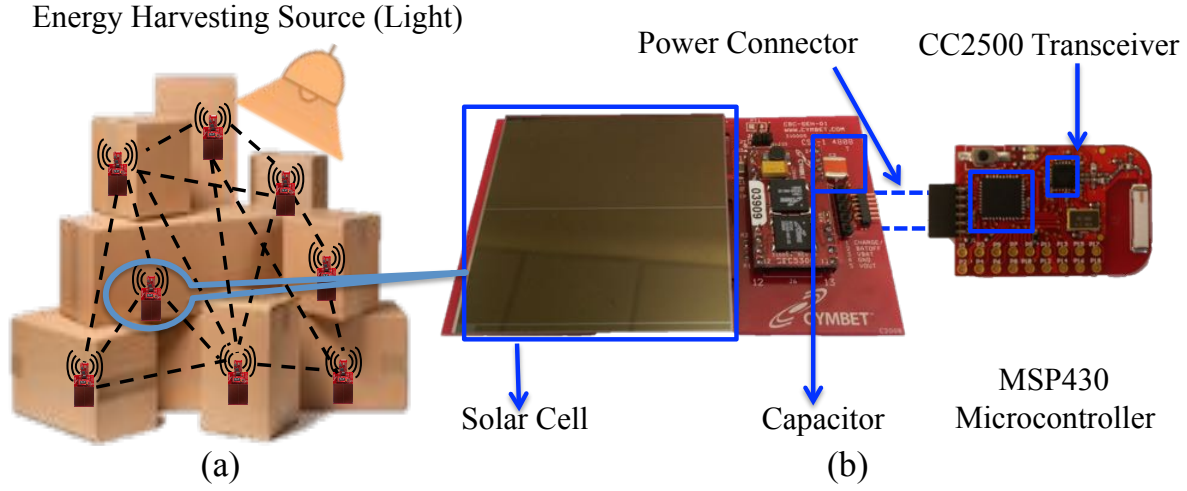


Figure 1.6: (a) EH nodes can be attached to everyday objects (e.g., boxes) and utilize neighbor discovery protocols for inventory tracking and monitoring applications. (b) The prototype EH node stores harvested energy in a capacitor and powers the microcontroller and transceiver.

1.2.1.2 Panda: Neighbor Discovery on a Power Harvesting Budget

In Chapter 3, we present Panda, a Neighbor Discovery (ND) protocol for Energy Harvesting (EH) nodes. Panda enables object tracking and monitoring applications in EH networks by *identifying* nodes which are within wireless communication range of one another. ND has always been an important part of many network scenarios [131, 148]. Yet, to consume power at the rate of power harvested, EH nodes require *extremely limited power budgets*: we show that, even with optimized power spending, the duty cycles are between 0.1-0.6%. Therefore, numerous assumptions from related works (e.g., [24, 149]) no longer hold, including that switching times (between the sleep, receive, and transmit states) draw negligible power and that the power costs to send and receive are identical. Furthermore, in the envisioned applications, the node's main task is to perform ND, and thus, the power consumed by ND is the dominant component of the power budget.

First, we consider a generic model of an EH node and study, for the first time, important properties of the radio in the context of ND. We then develop the Panda protocol in which an EH node discovers its neighbors by transitioning between the sleep, listen, and transmit states at rates that satisfy a power budget. Furthermore, we present Panda-Dynamic (Panda-

D), which extends Panda's applicability to non-homogeneous power harvesting and multihop topologies. Using techniques from renewal theory, we derive closed form expressions for the discovery rate and the power consumption. We develop the Panda Configuration Algorithm (PCA) to determine the node's duration in each state (sleep, receive, transmit), such that the discovery rate is maximized, while meeting the power budget. The solution obtained by the PCA is numerically shown to be between 94–99.9% of the optimal for all scenarios considered.

Using TI eZ430-RF2500-SEH EH nodes [5], we show that the real-life discovery rates are within 2% of the analytically predicted values, demonstrating the practicality of our model. Moreover, we show that Panda's *experimental* discovery rate is up to 3 times higher than the discovery rates from simulations of two of the previously best known low-power ND protocols [24, 115]. Furthermore, we demonstrate that Panda-D adjusts the rate of ND for scenarios with non-homogenous power harvesting and multihop topologies. Our study is motivated by the commercially-available EH nodes that compose our testbed. However, the applicability of our design extends to other platforms.

The main contributions of this work are to (i) model an EH node in the context of ND, and (ii) design, (iii) optimize, and (iv) experimentally evaluate Panda. These contributions have resulted in a provisional patent [111] and are pending publication in [110].

1.2.2 Exploiting Mobility in Proportionally Fair Cellular Scheduling

In Chapter 4, we present the Predictive Finite-horizon PF Scheduling ((PF)²S) Framework *which is tailored for mobile nodes and that takes advantage of both slow- and fast-fading*. To support the framework, we conducted an extensive 3G wireless channel measurement campaign, measuring the signal quality to each sector (E_c/I_o), in drives spanning 810km and during a period of over 1,300 minutes. We analyze the traces and show that mobile users experience pronounced slow-fading. We observe that the slow-fading component of E_c/I_o is remarkably reproducible for multiple drives on the same path (e.g., Fig. 1.3(b)). Based on this observation, we develop a rate prediction mechanism which is fed into the framework.

Using test cases generated from the collected traces, we perform an extensive simulation evaluation of the (PF)²S Framework. We show that various instances of the framework consis-

tently outperform the existing PF scheduler. Specifically, throughput improvements in realistic mobile scenarios range from 15% to 55% (with maintained or improved fairness levels).

The main contributions of this work are 3-fold: (i) it demonstrates, based on an extensive measurement campaign, that mobile users experience a reproducible but non-stationary slow-fading channel; (ii) it provides a cellular scheduling framework (and corresponding algorithms), tailored for mobile users; and (iii) it shows (using trace-based simulations) that the framework can significantly improve performance. We also describe these contributions in [113,114] as well as a pending patent application [13].

1.2.3 Performance Evaluation of Fragmented Structures

In Chapter 5, we study fragmentation within a probability model of the combinatorial structures. Although our model does not refer to any particular application, it is applicable to many practical systems including cognitive radio networks as listed above. As such, we use a general notation: a *resource* is composed of *slots* and represents, for example, the wireless spectrum, and an *item* represents, for example, a request for sub-channels.

Two intriguing questions are center stage: How bad can fragmentation become and at what rate does it develop? For example, can fragmentation progress to a point where nearly all items are *completely fragmented*, i.e., for every item size $j > 1$, are nearly all size- j items allocated j mutually disjoint slots?

The main contributions of this work are as follows. We define a probability model specifying distributions for item sizes and residence times, and then analyze the system operating at capacity. If the resource size is much larger than the maximum item size, then as the fragmentation process approaches equilibrium, nearly all of the allocated items are completely fragmented, i.e., the occupied units are mutually disjoint. In a suite of four theorems, we show how this result specializes for certain classes of item-size distributions. However, as a counterpoint to these rather intimidating results, we present the findings of extensive experiments which show that the delays in reaching the inefficient states of nearly complete fragmentation can be surprisingly long. We also describe these contributions in [37,40].

1.2.4 Project-based Learning

Appendix 6 describes our experience in engaging a large and diverse group of students in project-based learning within a large-scale interdisciplinary EnHANTs research effort. Our experience with organizing multiple student projects to contribute to a large-scale effort is unique and this chapter describes our approaches and some of the lessons we learned.

Over 17 semesters, we have involved a diverse population of 80 high school, undergraduate, Masters, and Ph.D. students in 180 semester-long research projects related to the design and development of the EnHANT prototypes and the prototype testbed. The student projects are multidisciplinary. A project typically focuses on one disciplinary area, but requires interaction with at least two other areas. The projects necessitate collaboration, provide students with in-depth fundamental understanding of networking concepts, and require students to improve their communication skills. We use “real-world” system integration deadlines and frequent system demonstrations to motivate students and to encourage cross-disciplinary collaboration. Students demonstrated prototypes and the testbed at six conference demonstration sessions [62, 66, 112, 145, 162, 179] and at over three dozen additional live on-site and off-site demonstrations. Students also contributed to publications describing the prototypes and the testbed [61, 109]. To evaluate our learning activities, we conducted a survey among the students. Of the students who completed the survey, over 90% indicated that the project was rewarding and enriching, and 70% indicated that working on this project improved their ability to function on multidisciplinary teams *more than any other activity in their academic career*. These contributions were also described in [65, 108].

This page intentionally left blank.

Chapter 2

Energy Harvesting Active Networked Tags (EnHANTs): Prototyping and Experimentation

This chapter focuses on a new type of wireless devices in the domain between RFIDs and sensor networks – Energy Harvesting Active Networked Tags (EnHANTs). Future EnHANTs will be small, flexible, and self-powered devices that can be attached to objects that are traditionally not networked (e.g., books, furniture, toys, produce, and clothing). Therefore, they will provide the infrastructure for various tracking applications and can serve as one of the enablers for the Internet of Things. We present the design considerations for the EnHANT prototypes, developed over the past 4 years. The prototypes harvest indoor light energy using custom organic solar cells, communicate and form multihop networks using ultra-low-power Ultra-Wideband Impulse Radio (UWB-IR) transceivers, and dynamically adapt their communications and networking patterns to the energy harvesting and battery states. We describe a small scale testbed that uniquely allows evaluating different algorithms with trace-based light energy inputs. Then, we experimentally evaluate the performance of different energy harvesting adaptive policies with organic solar cells and UWB-IR transceivers. Finally, we discuss the lessons learned during the prototype and testbed design process.

The design and development of the EnHANTs are a joint effort of several research groups. Our contributions are in the design and development of Medium Access Control (MAC) protocols, overall prototype integration, and the transceiver integration. Major design contributions were also made by Ph.D. candidates M. Gorlatova (energy harvesting adaptive algorithms), J. Sarik, and B. Vigraham (hardware). Versions of different prototype and testbed components were designed as part of undergraduate and M.S. student projects (see Appendix 6), as acknowledged via authorship in [62, 66, 145, 179].

Partial and preliminary versions of this Chapter appeared in demo descriptions [62, 66, 145, 179], in Proc. of IEEE INFOCOM'13 mini-conference [61], and in a pending journal submission [109].

2.1 Introduction

Energy Harvesting Active Networked Tags (EnHANTs) will be a new type of ultra-low-power flexible devices that harvest indoor light energy and can be attached to objects that are traditionally not networked. The realization of the EnHANTs vision is based on recent advances in energy harvesting [25, 123, 125, 151], which allows for dramatic reduction in the size and weight of a wireless node battery. In particular, advances in the area of organic semiconductors for energy harvesting allow the fabrication of organic photovoltaics¹ (OPVs) on flexible substrates [125], thereby enabling the pervasive use of future flexible EnHANTs. Another enabling technology is ultra-low-power Ultra-Wideband Impulse-Radio (UWB-IR) communications [44, 46, 165]. UWB-IR spends substantially less energy than other low-power wireless technologies due to its pulse-based nature.

The advances in these areas will allow small, flexible, lightweight tags to communicate and network in environments with limited energy availability (e.g., while harvesting indoor light energy).² The form factor for the envisioned EnHANT is shown in Fig. 1.2(a) and a recent

¹We use the term ‘photovoltaic’ rather than ‘solar cell’ to emphasize the use for harvesting *indoor light*, rather than sunlight.

²Indoors, the amount of energy available is a *thousand times lower* than outdoors [60].



Figure 2.1: A software-based light control system, along with 4 EnHANT prototypes.

mock-up is shown in Fig. 1.2(b). A tag of this size (less than 25cm^2) will support continuous data rates of nearly 3kbps [68].

EnHANTs will be one of the enablers for the *Internet of Things* [20] and will support a variety of tracking and monitoring applications beyond what the current tracking technology, RFIDs, permits. While RFIDs make it possible to *identify* an object, EnHANTs will make it possible to *search* for an object, and to continuously track objects' whereabouts, including their proximity to each other.

Realizing the EnHANTs vision requires the collaborative design of cross-layer hardware, software, and algorithms. A few solar (sunlight) powered sensor networks have been deployed and tested [151, 169]. Additionally, networking protocols for energy harvesting nodes recently started gaining attention (e.g., [34, 57, 102]). However, the challenges imposed by the cross-layer interactions between indoor energy harvesting, RF transceiver design, communications, and networking have not been studied in depth.

Developed over the past 4 years through a series of design phases, in this chapter we present the cross-layer design considerations of the EnHANT prototypes. The current prototypes (shown in Fig. 1.5) are much larger than the envisioned node that appears in Fig. 1.2(a). Yet, they already harvest indoor light energy using custom-designed OPVs (which can be made flexible) and communicate wirelessly using ultra-low-power UWB-IR transceivers. The prototypes form small networks and adapt their communications and networking patterns to the battery and energy harvesting states using an Energy Harvesting Module (EHM). The prototype design has been driven by the needs of the higher layers (e.g., flow control, multihopping), while the

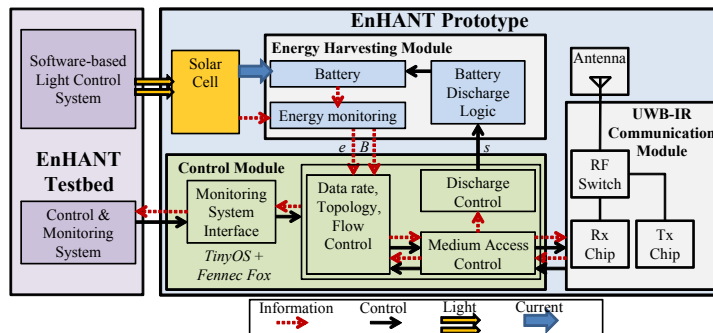


Figure 2.2: A block diagram of the EnHANT prototype, and its interactions with the testbed.

higher layer protocols (e.g., MAC and flow control) had to adapt to the physical layer (OPV and UWB-IR) characteristics.

Moreover, we present the EnHANTs testbed,³ which enables controllable and repeatable experiments with communications and networking algorithms for energy harvesting nodes. The testbed allows observing the states of the prototypes in real time and includes a software-based light control system (see Fig. 2.1) that can expose solar cells to controllable light conditions based on real-world light energy traces. To the best of our knowledge, this environmental input control system is the first of its kind. The approach to testbed design and development we pioneer in this work, as well as the design of the environmental energy inputs, is applicable not just to EnHANTs, but to all future energy harvesting devices.

Algorithms on all layers of the protocol stack need to be redesigned in order to support networking energy harvesting devices. However, due to space constraints, designing new energy harvesting adaptive algorithms is out of the scope of this chapter. Rather, using the testbed and the prototypes, we conducted experiments regarding the performance of the algorithms, some of which were proposed in [57,68,89]. Specifically, we considered: (i) algorithms that determine the energy spending rate of a node, (ii) the energy related tradeoffs of the UWB-IR transceivers in realistic environments, (iii) link layer algorithms that determine the data rates of communicating

³A video showing the testbed from the demonstration presented in [145] is available at youtu.be/QFCf621BATI

nodes, and (iv) flow control and collection tree selection algorithms for a small network. Our evaluation method is unique, since we provide the nodes various light energy scenarios and conduct repeatable experiments. Insights obtained in the experiments may apply beyond the EnHANTs to energy harvesting sensor networks [86, 132, 180] and networks of computational RFIDs [73]. Throughout the evaluation of experiments, we demonstrate numerous phenomena which are typically difficult to capture in simulations and nearly impossible to model in analytical work.

The prototypes and the testbed were developed over the past 4 years through a series of integration phases. At the end of each integration phase, the prototype and testbed functionality were demonstrated at a conference demonstration session (e.g., [62, 66, 145, 179]). We summarize the lessons that were learned throughout the different phases of the prototype and testbed design process. For example, we discuss gaps between theoretical assumptions and practical constraints as well as the importance of facilitating repeatable experiments in an energy harvesting testbed. We believe that these lessons would be useful for the designers of many different types of energy harvesters and energy harvesting adaptive networks.

The main contributions of this chapter are (i) the design and development of EnHANT prototypes, (ii) the design of a testbed for the evaluation of energy harvesting algorithms, and (iii) the experimental evaluation of such algorithms. To the best of our knowledge, the EnHANT prototypes are the first wireless devices that harvest energy using OPVs, the first to demonstrate multihop data forwarding over the ultra low power UWB-IR physical layer, and the first to adapt to the harvesting states in real time. We also note that this work is the first attempt to evaluate energy harvesting adaptive policies in a controllable experimental environment.

The chapter is organized as follows. In Section 2.2 we discuss the related work. Sections 2.3 presents an overview of the prototypes and the testbed. Section 2.4 introduces the notation and the experimental settings. Section 2.5 discusses the OPVs, the EHM, and energy allocation policies for energy harvesting nodes. Section 2.6 discusses the UWB-IR transceiver, the MAC layer, and the link layer energy harvesting adaptive policies. Section 2.7 discusses energy harvesting adaptive policies for networks of energy harvesting nodes. Section 2.8 presents the lessons learned and Section 2.9 presents ongoing and future work. We conclude the chapter in

Section 2.10.

2.2 Related Work

Combining the advances in energy harvesting and ultra low power communications has attracted attention from industry [2, 5] and academia. Energy harvesting can be used to power sensor nodes [5, 151], computational RFIDs [73, 177], and to realize mm³-scale wireless devices [3]. There are also energy harvesting sensors [2]; for example, NFC-WISPs [48] harvest RF-energy from a nearby smartphone to power an e-display. To the best of our knowledge, all of the prior platforms operate as part of a single-hop network.

The prototypes are integrated with a UWB-IR communication module which supports higher layer MAC and networking functionalities. An alternative approach for device communications is explored in [103]. There have been other low power UWB-IR implementations [46, 82, 144, 153, 165]. A few UWB-IR product development efforts target higher data rate applications such as wireless microphones [21], real time location systems [175], and radar [155]. However, to the best of our knowledge, none have implemented a network of more than two nodes, nor has any experimental MAC or higher layer communications been reported.

MAC protocols for UWB-IR based on a time-hopping physical layer (which differs from our transceiver) can be found in [45, 49, 90, 97, 166]. However, to the best of our knowledge, these MAC protocols have not been experimentally evaluated. For each of these protocols, carrier sensing, which is difficult to implement due to the low transmit power and the lack of a carrier waveform for UWB-IR communications [23, 176], is not utilized. This complicates the implementation of even simple CSMA schemes. Hence, the 802.15.4.a standard [81] provides a UWB-IR physical layer and an Aloha MAC layer with optional carrier sensing based on preamble detection [119].

The prototypes are integrated with a UWB-IR communication module which supports higher layer MAC and networking functionalities. An alternative approach for device communications is explored in [103]. There have been other low power UWB-IR implementations [46, 82, 144, 153, 165]. A few UWB-IR product development efforts target higher data rate applications such as wireless microphones [21], real time location systems [175], and radar [155]. However, to the

best of our knowledge, none have implemented a network of more than two nodes, nor has any experimental MAC or higher layer communications been reported.

MAC protocols for UWB-IR based on a time-hopping physical layer (which differs from our transceiver) can be found in [45, 49, 90, 97, 166]. However, to the best of our knowledge, these MAC protocols have not been experimentally evaluated. For each of these protocols, carrier sensing, which is difficult to implement due to the low transmit power and the lack of a carrier waveform for UWB-IR communications [23, 176], is not utilized. This complicates the implementation of even simple CSMA schemes. Hence, the 802.15.4.a standard [81] provides a UWB-IR physical layer and an Aloha MAC layer with optional carrier sensing based on preamble detection [119].

Using the prototypes and the testbed, we experiment with energy harvesting adaptive policies [57, 68, 89, 102, 121]. Many such policies were proposed for different layers of the networking stack [34, 57, 102, 117, 121, 157, 169, 180]. However, the majority of the proposed policies were only evaluated via simulations.

While several wireless network testbeds exist (e.g., [52, 151, 169]) and a few are under development (e.g., [47, 87]), to the best of our knowledge, our testbed is the first to support trace-based *light energy control* functionality. Solar simulators (used for testing solar cells) can provide precisely controlled illumination (approximating sunlight) but cannot create trace-based dynamic light environments.

2.3 Prototype and Testbed

In this section, we present an overview of the prototypes and the testbed used throughout the chapter for all experiments. Note that the version presented below corresponds to Phase V of the prototype and the testbed and we will describe its evolution in Section 2.8.1.

2.3.1 Prototype

The EnHANT prototype is shown in Fig. 1.5 and its block diagram, including the different modules and their interactions, is shown in Fig. 2.2. The current prototypes are larger than the envisioned nodes that appear in Fig. 1.2, yet they are already integrated with the enabling

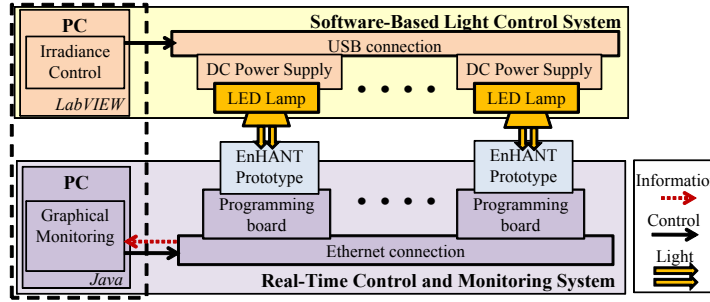


Figure 2.3: A schematic diagram of the EnHANTs testbed.

technologies.

The prototype’s Energy Harvesting Module (EHM) contains a rechargeable battery and energy monitoring circuitry. It interfaces with the solar cells, which are specifically designed to harvest *indoor light energy*. The current EHM is not powering the EnHANT prototype. The harvested energy is stored in the EHM’s battery, and the energy spending on communications and other activities is emulated by discharging a load.

The prototypes communicate with each other wirelessly using ultra-low-power Ultra-Wideband Impulse-Radio (UWB-IR) Communication Modules, based on UWB-IR transmitter and receiver chips (described in [44]). The custom chips are mounted onto a printed circuit board that interfaces with the other prototype components. A Complex Programmable Logic Device is used to realize the “glue logic” between the radio chipset and the rest of the prototype.⁴

The prototype Control Module (CM) is based on a legacy off-the-shelf MICA2 mote that runs TinyOS with an added Fennec Fox software framework [150]. The CM implements the MAC and forwarding protocols tailored for the UWB-IR transceivers. The Communication Module is integrated with the CM such that the packets, which contain node ID and network control information, originate in the TinyOS application layer and are sent *wirelessly via the UWB-IR transceiver*. The CM adapts the prototype’s networking and communication patterns, based on the energy states reported by the EHM.

⁴We note that while the design of the UWB-IR transceiver chips were described in [44], the integration of the transceiver with the higher layers, and evaluations in realistic environments (sending data packets, multi-user interference) have not been presented before.

Control	Agilent E3631A DC power supply with LabVIEW-controlled LED current
Lamps	Engin LZ4-00CW00 component LEDs
Enclosure and Mounting	Custom dark box with 3D printed solar cell mounting fixtures

Table 2.1: The hardware configuration for the light control system.

2.3.2 Testbed

Our small scale testbed, shown schematically in Fig. 2.3, includes 6 *EnHANT* prototypes, a *control and monitoring* system, and a *software-based light control system*. For control and monitoring, the prototypes are placed on MIB600 programming boards and accessed from a PC via Ethernet. On the PC, a Java-based graphical monitoring system shows in real time the network topology, data rates, energy harvested, battery levels, and the individual packets transmitted (they are shown as flashing ‘arrows’). A screenshot of the monitoring system is shown in Fig. 2.4.

The software-based light control system allows exposing individual prototypes to repeatable light energy conditions *based on real-world irradiance (light energy) traces*. The system is shown in Fig. 2.1, and the hardware configuration for the system is shown in Table 2.1. To ensure full control over the photovoltaics’ orientation and to eliminate external light sources, the photovoltaics are placed inside custom-designed dark box enclosures, shown in Fig. 2.5.⁵ The light sources are component cool white LEDs, mounted on heat sinks and attached to the enclosures. A LabVIEW script on a PC controls the irradiance inside each enclosure by controlling the current supplied to the LEDs from a DC power supply. The irradiance produced by the system was calibrated using a NIST-traceable photodiode (Newport UV-818). The system can produce over 3,000 distinct irradiance levels between 0 and 14mW/cm² (an effective resolution of less than 5μW/cm²); the irradiance levels can be changed with time steps of under 0.1sec.

⁵In practice, photovoltaics are effectively a connected series of modules; if a photovoltaic is partially covered, one of the modules will be darkened and the cell will produce significantly less power.

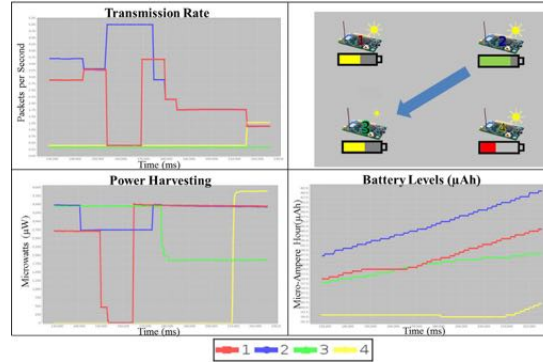


Figure 2.4: A screenshot of the EnHANT testbed monitoring system.

This system allows replicating real-world irradiance⁶ traces with remarkable repeatability. For example, energy harvesting rates over 4–10 repetitions of a light energy trace are shown in Figures 2.9(b), 2.13(a), 2.14(b), and 2.15(b), with variations in energy harvested less than 1.9%, 6.4%, 4.1%, and 3%, respectively. This ensures that experimental evaluations of energy harvesting adaptive policies are *based on the same energy inputs*.

2.4 Notation and Experimental Settings

The notation used in following sections is summarized below. We assume that time is slotted, and denote the time slot index by i . We denote the energy harvesting rate of node u in time slot i by $e_u(i)$, the desired energy spending rate by $s'_u(i)$, the actual energy spending rate by $s_u(i)$, and the data rate by $r_u(i)$. We denote the battery charge level of node u in slot i by $B_u(i)$. Prior to the experiments, the battery is conditioned to an initial charge point using a multimeter and a power supply. Throughout this chapter, $B_u(i)$ is reported in Coulombs.⁷ In determining node data rates, we consider the energy spent on transmitting and receiving a bit,

⁶Additionally, the system is capable of mapping real-world energy harvesting measurements from other sources (e.g., vibration [64]) to equivalent irradiance values to enable experiments based on other energy harvesting sources.

⁷Battery levels and battery capacities are often reported in various different units, units of charge (Coulombs, Ampere-hours), and units of energy (Joules). Since $1C = 1A \cdot 1sec$, the conversion between Coulombs and Ampere-hours is straightforward. To calculate the energy, the charge is multiplied by the voltage of the battery, V_{BAT} . In our system, $V_{BAT} = 2.4V$, thus the charge of $1C$ (or, equivalently, $278\mu Ah$) corresponds to $2.4J$ of energy.

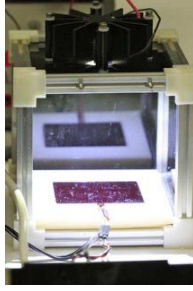


Figure 2.5: A dark box enclosure used in the light control system.

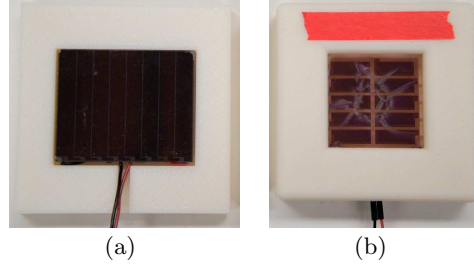


Figure 2.6: Solar cells integrated with EnHANT prototypes: (a) an amorphous silicon (a-Si) solar cell, and (b) a custom-fabricated organic photovoltaic (OPV).

denoted by c_{tx} and c_{rx} , respectively. The c_{tx} and c_{rx} for the current prototype transceiver/MAC are discussed in Section 2.6.3.

We experimentally evaluate several energy harvesting adaptive policies. For these evaluations, we provide the prototypes light inputs based on indoor light energy traces available in [71] using the trace identifiers specified therein (e.g., L-1, L-2, etc). The light control system allows ‘replaying’ light energy traces at different time and light intensity scales. In the experimental evaluations we ‘compress’ traces in time, downsampling original day-long traces to ‘play’ in 6 – 12 minutes. To capture the corresponding dynamics in energy storage behavior, we additionally scale the traces by a factor indicated in the corresponding sections. The scale-factor is chosen such that the irradiance of the light trace falls within the range of the light control system of 0-14mW/cm².⁸ In some sections, we compare experimental results to simulations, conducted in MATLAB.

2.5 Energy Harvesting

In the subsections below, we briefly describe the photovoltaics and the EHM, and then evaluate several energy allocation policies for energy harvesting nodes.

⁸In some cases, we do not scale the irradiance traces to the maximum light level of 14mW/cm².

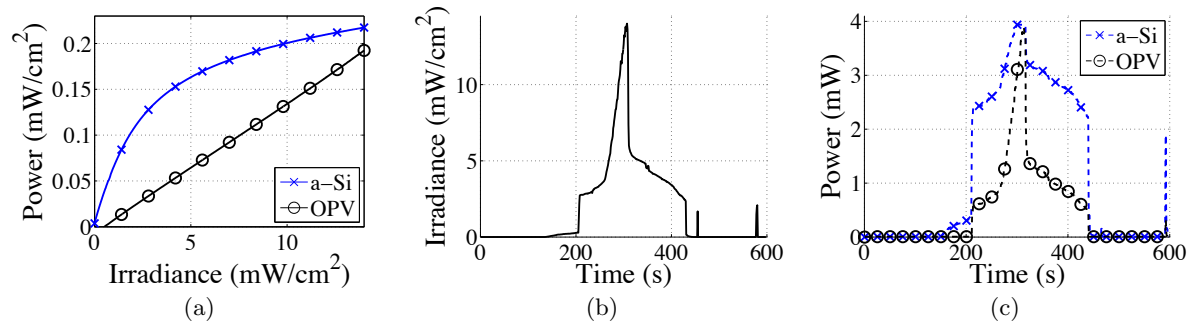


Figure 2.7: Energy harvesting for a prototype with an a-Si solar cell and an organic photovoltaic (OPV): (a) generated power as a function of irradiance (light intensity), (b) time-varying irradiance based on a light energy trace from [71], (c) power harvested by an a-Si solar cell and an OPV.

2.5.1 Photovoltaics

The prototypes are integrated with two different photovoltaic technologies (shown in Fig. 2.6): commercially available amorphous silicon (a-Si) cells, and *custom-designed polymer OPVs*.⁹ The a-Si cells are commonly used for indoor harvesting applications [5,172]. This is due to their higher efficiency over the conventional monocrystalline silicon cells under the reduced intensity and blue-shifted spectrum of indoor fluorescent or LED lighting. The OPVs have energy harvesting characteristics that are similar to those of the a-Si cells under indoor light, and they can be manufactured on *flexible substrates* using low temperature, roll-to-roll processing techniques.

Different types of photovoltaics perform differently under the same light conditions. For example, Fig. 2.7(a) shows the power generated by two different photovoltaics: an a-Si cell and an OPV (both biased at 3.1V, the minimum voltage required to charge the prototype's battery), under different irradiance (light intensity) levels. Using the EHM (described below), we measure the power produced by the cells. Correspondingly, using the software controlled lights, we can estimate the energy incident on the solar cells. The *harvesting efficiency* is defined as the ratio between the two. Hence we obtain the harvesting efficiency of the OPVs is 1%, while the harvesting efficiency of a-Si cells varies between 1% and 3% depending on the irradiance (see the

⁹The a-Si cell is Sanyo AM-1815 with a 5.61x4.52cm² active area. The OPVs area is 5.0x5.0cm². A description of the process we followed to fabricate the OPVs can be found in [128].

R_{LOAD}	R_{SENSE1}	R_{SENSE2}	τ	Max. avg. current	Resolution, instant. current	Resolution, battery charge
220Ω	10Ω	10Ω	20ms	5.12mA	625nA	$<5mC$

Table 2.2: EHM Configuration

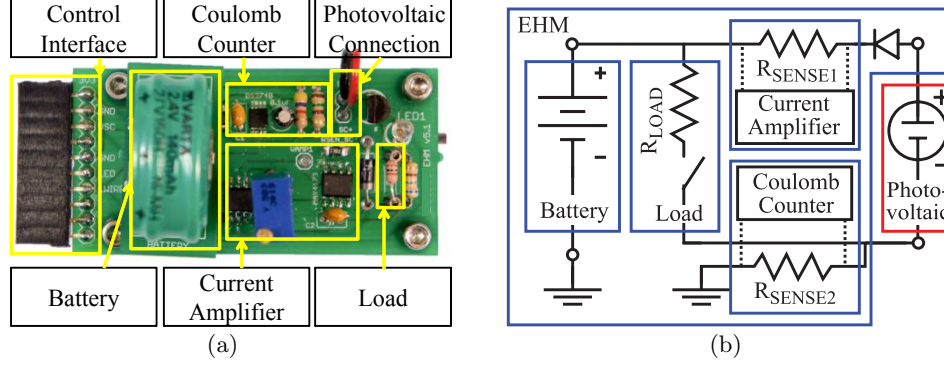


Figure 2.8: (a) A photo of the Energy Harvesting Module (EHM), and (b) its block diagram.

measurement results in Fig. 2.7(a)). Correspondingly, as shown in Fig. 2.7(c), when we expose the two photovoltaics to irradiance levels based on a light energy trace recorded over a day in L-1 (Fig. 2.7(b)), the ‘curves’ of the power generated by the two photovoltaics have different shapes. We note that these effects are difficult to capture in simulations (e.g., [57, 68, 78]), which simply assume that the energy harvested by a solar cell is a linear function of the irradiance.

2.5.2 Energy Harvesting Module (EHM)

The EHM stores the energy harvested by the photovoltaic in a rechargeable battery. The EHM is designed such that the photovoltaic or the battery can be easily added or removed; this feature is used to condition the batteries and to interchange OPVs with a-Si cells.¹⁰ A block diagram and a photo of the EHM are shown in Fig. 2.8. The EHM’s interactions with the other prototype

¹⁰The harvesting point of the photovoltaic is fixed to 3.1V, which is the minimum voltage required to charge the prototype’s battery. No power point tracking techniques (i.e., MPPT) are used. With this simple design, we minimize inefficiencies and enable a seamless interchange between the two employed photovoltaics (a-Si and OPVs).

components are shown in Fig. 2.2. The hardware parameters for the EHM are summarized in Table 2.2.

For energy storage, the EHM uses a 2.4V NiMH battery with 150mAh (540C, 1.3kJ) capacity. The battery is intentionally oversized (for the envisioned EnHANTs the battery capacity will be on the order of 1-2C) to allow emulating a wide range of different battery capacities in software. The EHM monitors, in near real time, the battery level, $B(i)$, and the energy harvesting rate, $e(i)$, and reports them to the Control Module. To track $B(i)$, the EHM's energy monitoring circuitry uses a Maxim DS2740 *Coulomb counter*, which measures and integrates the bidirectional current across R_{SENSE2} , hence tracking the changes to the battery level in Coulombs. To track $e(i)$, the EHM uses a Maxim MAX4173 high side current sense amplifier, which measures the instantaneous current across R_{SENSE1} . The Coulomb counter updates the battery level every 0.875s, and the battery level resolution is under 5mC.

The EHM does not supply energy to the other EnHANT prototype components. Rather, as shown schematically in Fig. 2.2, the EHM implements controlled energy spending functionality that corresponds to transceiver energy spending on transmitting and receiving packets.¹¹ The Control Module signals to the EHM to activate a small load, which spends energy at a requested rate $s(i)$. Removing the constraint of running the prototype using harvested energy allows us to experiment with various hardware and protocol configurations. In the EHM, the energy is spent by discharging the battery through the load resistor R_{LOAD} (see Fig. 2.8) for τ ms. Each load activation reduces the battery by $\Delta B = \tau \cdot V_{\text{BAT}} / (R_{\text{LOAD}} + R_{\text{SENSE1}}) = 208.68\mu\text{C}$. To verify the precision of the EHM's controllable energy spending, we compared the energy spending rates calculated according to this formula, and the energy spending rates we experimentally obtained, for a set of EHM load activation rates. For up to 12 load activations per second, the discrepancy was under 2.1%.

In a typical environment, an indoor energy harvesting node would have access to 1-12 J/cm²/day [68], therefore a device with a 5x5cm solar cell of 1% efficiency will harvest 0.25 - 3 J/day (corresponding to an average harvested power of 2.89 μ W-34.7 μ W). Thus, the cur-

¹¹In the envisioned EnHANT, the transceiver power consumption is the main source of energy consumed, and hence, the energy consumption from computation is ignored.

rently used battery can be used to emulate energy storage sizes up to 432x incoming energy.

In this work, we only consider energy harvesting from light as it is convenient to control, measure, and reproduce for experiments. However, we note that in practice, other sources may also be available in indoor environments (e.g., ambient RF [103] and kinetic [64]). Although not directly applicable to other energy harvesting sources, our EHM design can be used as a baseline for other energy harvesting sources.

2.5.3 Node Energy Allocation Policies

Many different *energy allocation policies* for energy harvesting nodes have been recently proposed [57, 89, 121]. We evaluate a set of simple policies that aim to achieve *energy neutrality* – full, yet not excessive, spending of the energy harvested by the nodes. These policies, briefly described below, closely match node energy spending rates, $s(i)$, to node energy harvesting rates, $e(i)$. In the following sections, we use these policies as building blocks for wider-scale (i.e., link and network) energy harvesting adaptive policies.

Exponential Policies (EX): The desired energy spending rate $s'(i)$ is set to the *exponential average* of the energy harvesting rate: $s'(i) \leftarrow \hat{e}(i) = \alpha \cdot \hat{e}(i-1) + (1-\alpha) \cdot e(i)$, $0 \leq \alpha \leq 1$. Similar policies were evaluated, via simulations, in [102].

Energy Profile-based Policies (EP- K): A node creates an *energy profile* $\{\bar{e}(1), \dots, \bar{e}(K)\}$ by determining its *expected harvesting rate*, $\bar{e}(i)$, for different time intervals i . Energy profiles are used as inputs to many proposed policies [34, 57, 68, 89, 102, 121]. In the EP- K policies (K corresponds to the number of time intervals), the node's desired energy spending rates are set to the expected energy harvesting rates: $s'(i) \leftarrow \bar{e}(i) \forall i \in K$. For example, the EP-1 policy (examined, via simulations, in [57, 68]), corresponds to a node spending energy at its *average expected harvesting rate* over the entire planning horizon.

We conducted extensive experiments with different EX and EP- K policies, providing nodes, equipped with either an a-Si solar cell or an OPV, dynamic energy trace-based light energy inputs. Examples of energy spending rates, recorded in prototypes running these policies, are shown in Fig. 2.9(c). For these experiments, we provided the prototype with the light input

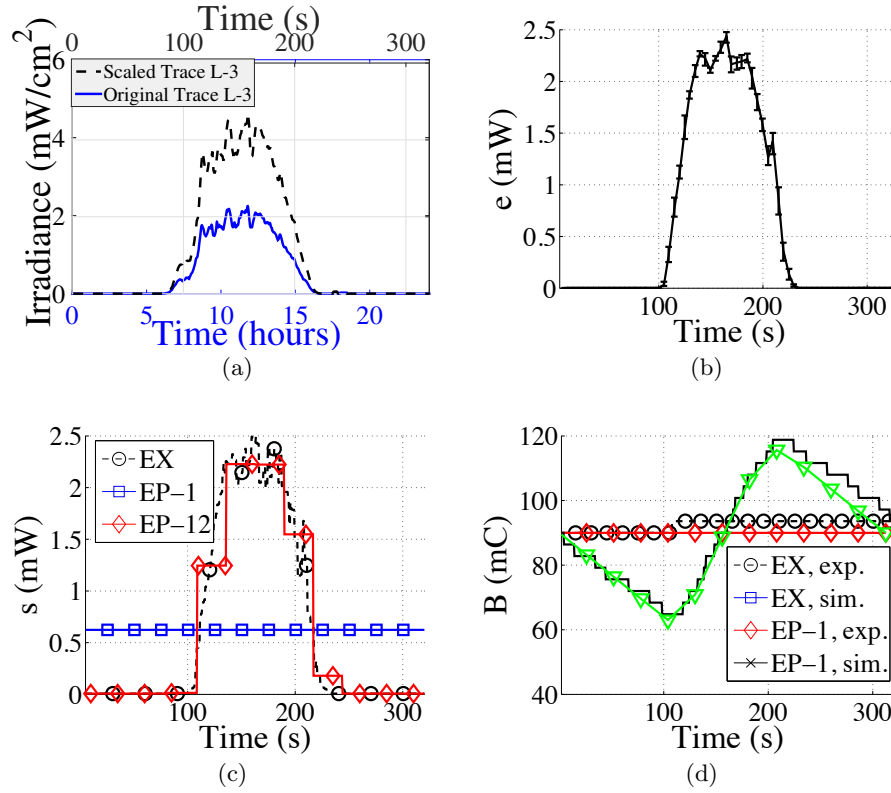


Figure 2.9: Node energy allocation policies: (a) the day-long irradiance trace from L-3 [71] compressed to 321s and scaled by 2.1x, (b) energy harvesting rates from 4 experiment repetitions with error-bars, (c) energy spending rates under the EX, EP-1, and EP-12 policies, and (d) battery levels, experimentally measured and simulated, under the EX and EP-1 policies.

shown in Fig. 2.9(a) corresponding to the light energy recorded over a day in L-3, compressed to 321s and scaled by 2.1x. The prototype was equipped with an a-Si solar cell. The energy spending rates are obtained for node energy harvesting rates illustrated in Fig. 2.9(b), where errorbars represent variations in energy harvesting rates in different experiments (the variability in total energy harvested was *under 1.9%*).

The experimental results and the results obtained via MATLAB simulations closely match. For example, for the policies shown in Fig. 2.9(d), the largest mismatch between the experimentally recorded and simulated battery levels was 5mC (one battery resolution level of the EHM). This confirms the reliability and precision of the energy state monitoring and controlled energy

spending functionalities of the EHM.

The EX and EP- K policies effectively ensure energy neutrality. For example, in all our experiments with the EX and EP- K policies corresponding to node energy harvesting rates shown in Fig. 2.9(b), the node used 95%-96% of the harvested energy. To the best of our knowledge, achieving such a level of energy neutrality is not feasible with currently available COTS devices (e.g., [5]). However, as we will demonstrate in subsequent sections, while these simple policies are effective ‘in isolation’, they do not take into account considerations important for *networked* nodes (e.g., that the nodes communicate, and thus may spend energy, in all time slots, including those with $s'(i) = 0$).

2.6 UWB-IR Communications, Medium Access Control, and Link Layer

The EnHANT prototypes communicate using ultra-low-power Ultra-Wideband Impulse Radio (UWB-IR) transceivers. The UWB-IR techniques used in the prototypes are fundamentally different from traditional narrow-band techniques. In UWB-IR, information is transmitted using very short *pulses* and most of the transceiver circuitry can be shut down between the pulses, resulting in significant power savings [160]. Yet, pulse-based communications pose many challenges. Clear Channel Assessment (CCA) functionality, for example, is not straightforward in UWB-IR. This is because, in UWB-IR there is no continuous carrier waveform to sense and the low transmit power complicates the discerning of a busy channel from narrowband interference [176]. Additionally, in UWB-IR the energy required to receive a bit is an order of magnitude higher than the energy required to transmit a bit [44, 165].

2.6.1 UWB-IR Chipset and EnHANT Prototype Integration

The UWB-IR transceiver [44] operates in a 500MHz band around a center frequency of 3.8GHz. The modulation scheme is Synchronized On-Off Keying (S-OOK), shown in Fig. 2.10, which permits low power timing acquisition. An S-OOK symbol consists of a synchronization pulse, which is always present, and a data pulse, which is only present for a ‘1’ bit. These pulses

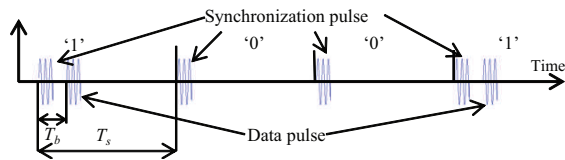


Figure 2.10: S-OOK modulation and parameters.

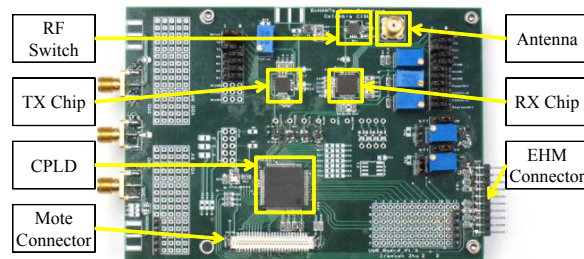


Figure 2.11: The UWB-IR Communication Module.

are separated by a time interval T_b . Adjacent symbols are separated by a time interval T_s . Targeting low data rate applications, we currently use $T_b = 1\mu s$, $T_s = 54\mu s$, for a transmission rate of 18.5kbps.

The UWB-IR Communication Module, shown in Fig. 1.5 and represented schematically in Fig. 2.2 (as part of the EnHANT prototype diagram), is implemented on a dedicated printed circuit board (see Fig. 2.11). The board integrates the transceiver chipset and a $1.5\text{cm} \times 1.2\text{cm}$ omnidirectional UWB antenna [107] (shown in Fig. 1.5) with the EnHANT prototype. A wide-band RF switch alternately connects the transmitter and receiver to the antenna port, enabling half duplex communication.

To integrate the UWB-IR transceiver with the software protocol stack, we developed a device driver under the TinyOS and the Fennec Fox framework [150]. This framework provides a layered modular programming environment, enabling separate interchanging of software modules at different protocol stack levels, and thus allowing operating over a custom physical layer. The driver provides an abstract interface to the MAC layer, and functionalities such as transmit/receive mode switch, packet serialization, buffering, and Clear Channel Assessment (CCA).

*Previous work on MAC for UWB-IR assumes that performing CCA is not possible. Yet, for the non-time-hopping UWB-IR communications used in our prototypes, we managed to implement CCA using a simple threshold-based energy detector.*¹² The receiver chip listens

¹²In *time hopping UWB-IR* [166], pulses are transmitted at different time slots according to a time hopping code. Due to the low required data rates of our applications, our UWB-IR transceiver foregoes the added complexity

Cell Phone Distance	0m	0.25m	0.50m	0.75m
Packet Error (%)	26.34%	23.64%	3.44%	0.44%

Table 2.3: The packet error rates when a cell phone placed calls at varying distances from the receiver.

for a pulse for $2 \cdot T_b$; if no pulse is demodulated, the channel is declared idle. Energy spikes from interfering systems [22] may cause CCA to falsely report a busy channel. However, for our purposes, these CCA ‘false alarms’ do little to affect our application which can wait until the noise spikes dissipate to transmit a packet. In future work we can further reduce this interference by adding a dedicated UWB filter that rejects energy from known frequency bands (i.e., cellular, WiFi).

2.6.2 Single Link Experiments

Prior to performing MAC experiments, we extensively characterized the UWB-IR communication module’s sensitivity to narrowband interference, transmission-rate limits, and packet-error rates under different conditions.

The UWB-IR frequency spectrum overlaps many frequency bands from narrowband communications systems including, for example, cellular and WiFi frequencies. Therefore, we tested the transceiver’s vulnerability to a given narrowband interfering system by placing a smartphone at varying distances from the receiver and generating a 5 minute call at each distance. The packet error rates are shown in Table 2.3. The results indicate that our transceiver is susceptible to narrowband interference at distances less than 0.5m which is sufficient for our experiments. Cellular interference is a representative example of narrowband systems in our testing environment, and therefore, we generalize these results and conclude that the UWB-IR transceiver and the CCA scheme operate uninhibited when interfering transmitters are kept at least a couple of meters away.

While the transceiver is capable of a transmission rate up to 18.5kbps, computational con-

of time-hopping, and as a result, only one node accesses the channel at a time. On the other hand, this enables the implementation of the described CCA scheme.

straints limit the effective transmission rate when sending packetized data.¹³ Note that the computational constraints are not fundamental, but are a limitation of our current prototype implementation and microcontroller. We therefore conducted experiments to characterize the effective transmission rate limit for a single link. In a first set of experiments, 3,000 packets were sent over a link in one direction, for varying packet rates and a constant packet size.¹⁴ In each experiment (i.e., for each packet size and rate), the receiver reported how many packets were correctly received. In a follow up experiment, the receiver logged information on packet arrival time by sending data via Ethernet to the prototype monitoring system.¹⁵

The results of these experiments are shown in Fig. 2.12(a). For a fixed packet size of 14 bytes, a 99% packet success rate was achieved for effective transmission rates less than 6.2kbps and 8kbps with and without logging, respectively. After exceeding these rates, packets always fail. This is due to interrupt overhead incurred mainly from packet transmission and reception. Increasing the packet length to 50 bytes enables higher transmission rates. This is due to the lower computational load resulting from additional time between successive packet reception interrupts. In the following sections, by using a 50 byte packet with effective transmission rates not exceeding 5kbps, we minimize the packet errors caused by computational overloading of the system.

We characterized the effect of varying the packet length and distance between nodes on the packet error rate. The results, shown in Fig. 2.12(b) indicate that the nodes achieve a very low packet error rate at short distances even for long packets. At the maximum packet length of 620 bytes, the effect of the added distances is observed. Therefore, for all experiments reported below, we minimize the packet error rates by setting the distance between nodes to 0.17m.

¹³We convert to effective transmission rate = (packet length)*(packet rate).

¹⁴Unless otherwise stated, nodes were directly adjacent to each other at a distance of 0.17m.

¹⁵Sending information through the Ethernet port is imperative for many of our experimental results. Therefore, it is important to characterize the system performance with this constraint.

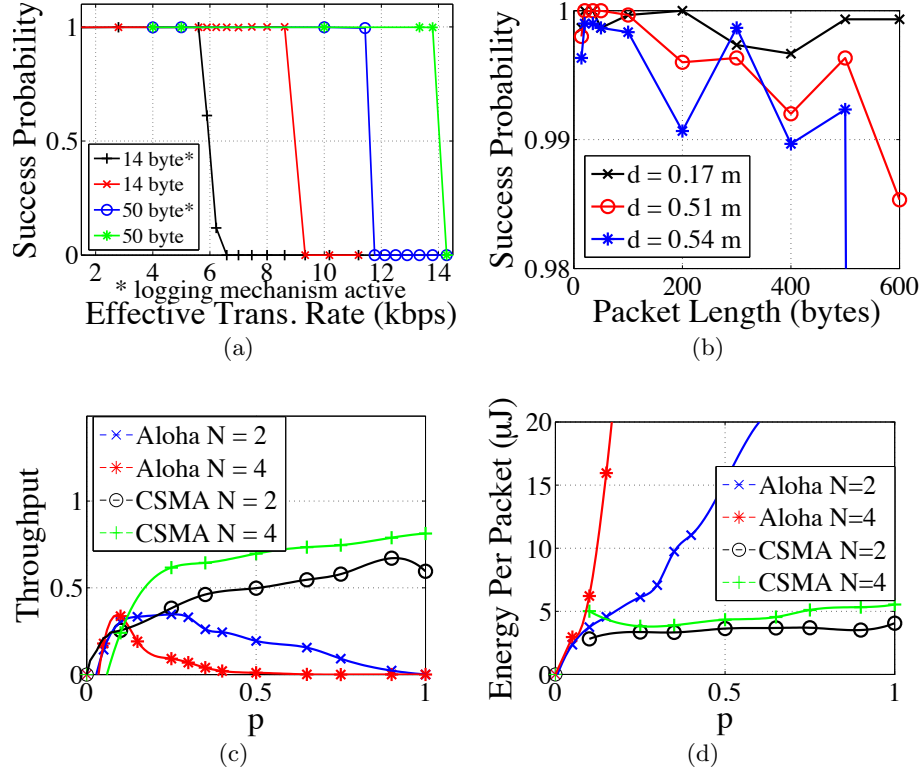


Figure 2.12: (a) Packet success probability as a function of packet rate for 14 and 50 byte packets with logging enabled and disabled, (b) packet success probability as a function of packet length and distance between nodes, (c) average throughput as a function of p , the transmit probability for Aloha and CSMA (100% throughput is 18.5kbps), and (d) average energy spent per successful packet transmission, as a function of p .

2.6.3 Medium Access Control (MAC) Evaluation

On top of the UWB-IR physical layer, we implemented two basic MAC protocols: p -persistent CSMA (enabled by our implementation of CCA) and the slotted Aloha.¹⁶ To the best of our knowledge, while more sophisticated protocols have been theoretically analyzed, this is *the first attempt to implement and experimentally demonstrate MAC protocols over a UWB-IR physical layer*.

¹⁶In Aloha, given a packet, a node transmits in a slot with probability p . In p -persistent CSMA, with probability p , a node senses the channel, and transmits if the channel is determined to be idle.

CCA requires listening to the channel, which is a relatively energetically expensive operation in UWB-IR. Thus, we conducted experiments to analyze the tradeoff between the added energy consumption of CSMA and the increased throughput.

For both MAC protocols, Figures 2.12(c) and 2.12(d) show experimentally obtained throughput and energy spending values, correspondingly. The evaluations were conducted in networks of 2 and 4 ‘infinitely backlogged’ (i.e., sending as often as possible) prototypes. Energy spending per successful packet, shown in Fig. 2.12(d), was calculated based on transceiver energy consumption data provided in [44] (0.31mW transmitting, 3.9mW idle listening). Clearly, the Aloha throughput ‘curves’, shown in Fig. 2.12(c), bear strong resemblance to classical analytical Aloha results.

Expectedly, in terms of throughput, CSMA outperforms Aloha. Additionally, despite the added cost of CCA, the collision avoidance provided by CSMA reduces the average energetic cost per successful packet significantly as compared to Aloha. Expectedly, CSMA throughput increases as p increases. Yet, CSMA throughput is lower than the theoretical values, due to a delay between sensing a free channel and sending a packet, and due to delays imposed by the prototype microcontroller (whose resources are occupied by other processes, i.e., energy monitoring). In all subsequently reported experiments, the prototypes use the *1-persistent CSMA MAC*.

2.6.4 Energy Costs

To demonstrate energy adaptive networking, we must first quantify c_{tx} and c_{rx} , the expected energy spent (at the MAC and physical layers) to send and receive a bit, respectively. Using 1-persistent CSMA, c_{tx} and c_{rx} will increase as a function of channel utilization. Yet, with some approximations, we estimate c_{tx} and c_{rx} for a 50 byte packet using power consumption data from [44] as follows. Transmitting a packet consumes $6.7\mu\text{J}$. An additional $0.41\mu\text{J}$ is consumed per CCA. If perfectly synchronized, the receiver would consume $31.3\mu\text{J}$ per packet reception. However, the receiver must idly listen for the start of the packet which will vary based on clock drift. The current prototypes rely on the microcontrollers of MICA2 motes (ATmega 128L AVR); their internal clock drift is rated at 3%. Assuming a packet transmission every 200ms,

we set the guard time to $\pm 6\text{ms}$ which results in $45.6\mu\text{J}$ per guard time. Further, the receiver must listen for 1 guard time around every possible CCA slot until it hears the packet. We therefore limit the sender to performing 1 CCA and if the channel is busy, drop the packet. This bounds the maximum sending and receiving per packet at $7.1\mu\text{J}$ and $76.9\mu\text{J}$, respectively, and we therefore estimate $c_{\text{tx}} = 17.8\text{nJ/bit}$ and $c_{\text{rx}} = 179.8\text{nJ/bit}$.

Due to our conservative estimations regarding the additional overheads, these c_{tx} and c_{rx} values are one or two orders of magnitude higher than the chipset energy consumption (1nJ/bit transmitting and $2\text{-}3\text{nJ/bit}$ receiving [44]); however, even under the conservative assumptions, the energy consumption is lower than that of Zigbee (296.4nJ/bit transmitting and 324nJ/bit receiving [100]), Bluetooth Low Energy (153nJ/bit [143]), or the CC2500 radio from [5] (200nJ/bit). Aligning the overall energy consumption with that of the chipset is a subject for future work. Note that it is significantly more expensive to receive than it is to transmit, with the majority of the energy consumed when searching for the start of the packet. This can be improved by using a clock with higher accuracy, using clock synchronization techniques (including techniques based on properties of *indoor lighting* [101]), or through receiver initiated MAC protocols such as [51].

2.6.5 Energy Harvesting Aware Data Rate Adaptation Policies

Communicating energy harvesting nodes need to jointly adapt their link layer behavior. Hence, we experimentally evaluated energy-harvesting-aware link data rate adaptation policies – the *Decoupled Rate Control* (DRC) policies, proposed in [68]. These policies aim to achieve *energy neutrality* (see Section 2.5.3). Under the DRC policy, prototypes u and v determine, independently, their desired energy spending rates $s'(i)$, using the EX or EP- K policies described in Section 2.5.3. The prototypes exchange these values, and calculate their data rates as $\{r_u(i), r_v(i)\} \leftarrow \max\{\min[s'_u(i), s'_v(i)]/(c_{\text{tx}} + c_{\text{rx}}), r_{\min}\}$, where r_{\min} is the minimum rate required to maintain a link.

We extensively evaluated the DRC policies with different light energy inputs. Fig. 2.13(b), for example, shows the data rates assigned by the DRC policy, in combination with the EX and EP-1 policies, corresponding to energy harvesting rates shown in Fig. 2.13(a) (where errorbars

represent variations in energy harvesting rates in different experiments). For these experiments, we provided the prototypes with light inputs corresponding to the light energy recorded over a day in L-1 and L-2 (nearby locations in the same office), compressed to 700s and scaled by 20.0x. The prototypes were equipped with OPVs. In 10 experiments with these light inputs, the variability in the total energy harvested was under 4.7% for node u , and under 6.1% for node v .

The experimentally obtained link data rates are slightly lower than the data rates obtained in simulations. For example, Fig. 2.13(c) shows experiments with the DRC policy in combination with EP-12 (each data point in Fig. 2.13(c) and 2.13(d) corresponds to a complete EnHANT testbed experiment that lasts 700 seconds). For node energy harvesting rates shown in Fig. 2.13(a), nodes u and v achieved, correspondingly, 90–92% and 84–87% of the data rates expected based on the simulations. These discrepancies are attributed to packet errors, which are not captured by the simulations, and are the result of MAC layer collisions, system computational overloads, and noise-induced bit errors.

In combination with the EP-1 policy, the DRC policy achieves energy neutrality. Combined with other policies, it may, due to its *per time slot* calculations, underspend the overall energy harvested. Moreover, it may also *overspend energy* if nodes need to communicate (e.g., to maintain synchronization, or to exchange basic energy awareness information) in all time slots, including those in which $\min\{s'_u(i), s'_v(i)\} = 0$. Both underspending and overspending of energy can be observed in Fig. 2.13(d), which shows the energy spent by nodes running the DRC policy in combination with the EX and EP-12 policies, for energy harvesting rates shown in Fig. 2.13(a).

2.7 Energy Harvesting Adaptive Flow Control and Tree Selection

In this section, we evaluate, using a small multihop network of prototypes, policies for networks of energy harvesting nodes. To the best of our knowledge, the EnHANT testbed network is the first UWB-IR-based multihop network. We note that most algorithms proposed for networks of energy harvesting nodes [34, 78, 102] are too complex for implementation in ultra-low-power indoor environments (recall that indoors the amount of light energy available is a thousand times

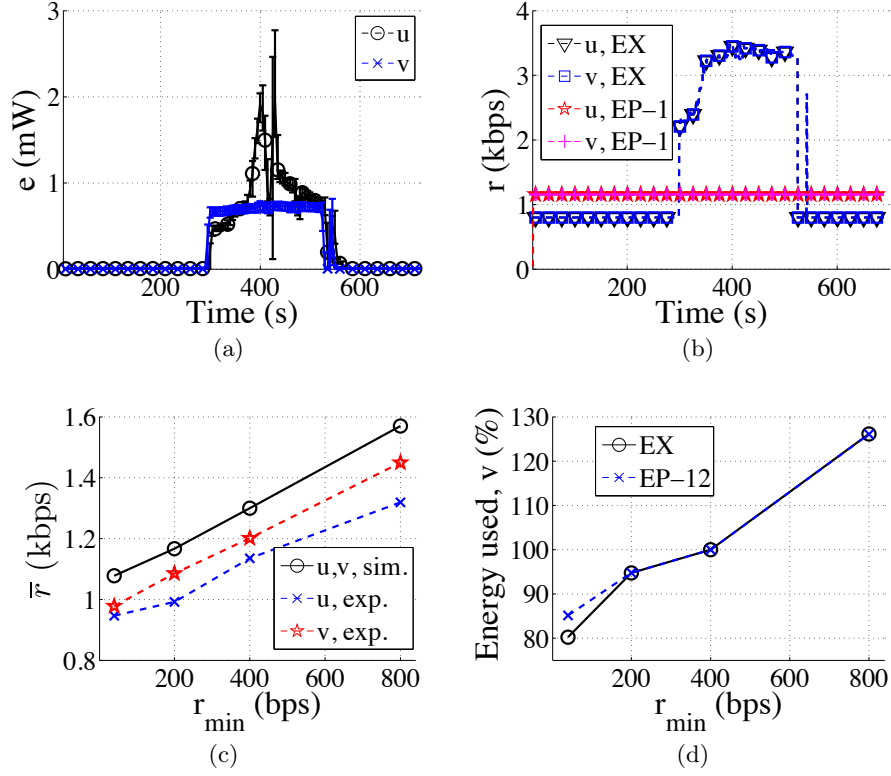


Figure 2.13: Decoupled Rate Control (DRC) link policies: Using a day-long irradiance trace compressed to 700s and scaled by 20.0x from L-2, L-3, and the corresponding (a) energy harvesting rates from 10 experiment repetitions with errorbars, (b) data rates under the EX and EP-1 policies, (c) average data rates under the EP-12 policy, experimentally obtained and simulated, and (d) percentage of harvested energy used by v , under the EX and EP-12 policies.

lower than outdoors), as they require multiple local [34] or global [102] iterations, or complex calculations [78]. We thus focus on evaluating simple policies and heuristics.

Our evaluations focus on data collection scenarios (e.g., ID collection), corresponding to the envisioned object tracking applications. Figures 2.14(a) and 2.15(a) show the considered network topologies: a *3-node line* network and a *diamond* network. In these topologies, prototypes u , v , and w generate messages, and send them, via multihop collection trees, to a prototype that serves as a *Collection Coordinator* (CC). In Section 2.7.1, we evaluate a set of flow control policies that assign data rates to network nodes. In Section 2.7.2, we briefly evaluate heuristics

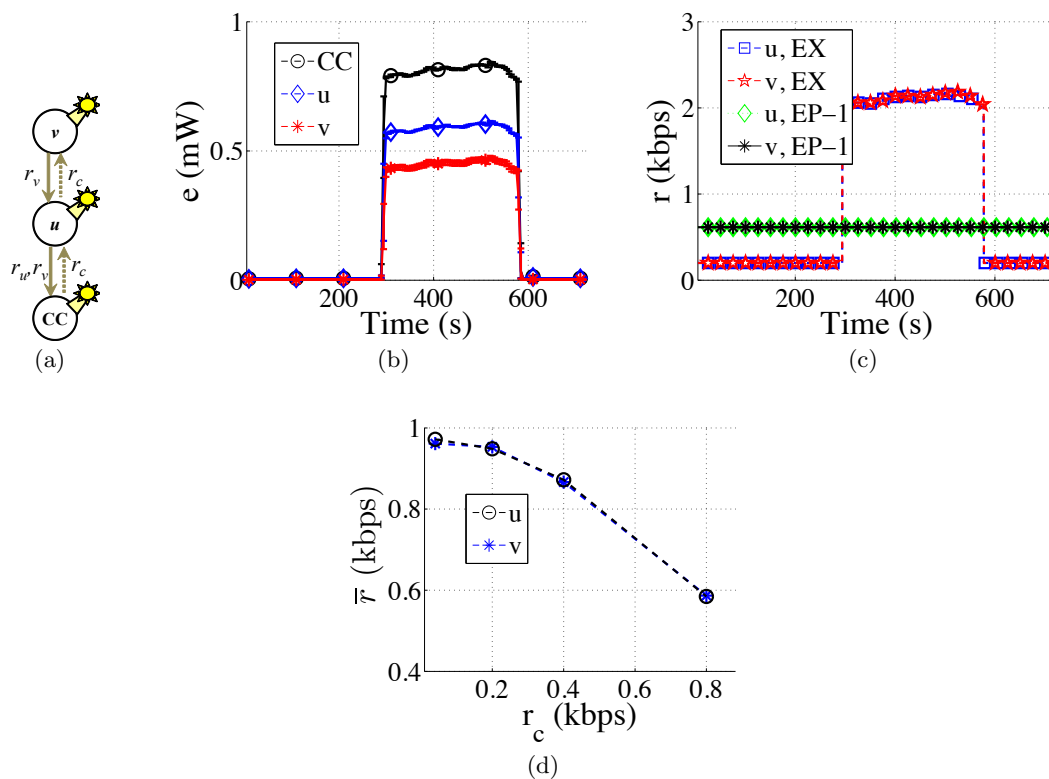


Figure 2.14: FLEX flow control policies for (a) a 3-node line network: Using a day-long irradiance trace compressed to 720s and scaled by 60.4x from L-2, and the corresponding (b) energy harvesting rates from 8 experiment repetitions with errorbars, (c) data rates under the EX and EP-1 node energy allocation policies, and (d) average data rates, under the EX policy, for different values of r_c .

for energy harvesting adaptive collection tree selection.

2.7.1 Flow Control

Our experiments with networks of energy harvesting nodes strongly indicated the need for flow control policies. When nodes set their data rates without considering other nodes (i.e., by setting $r(i) \leftarrow s'(i)/c_{tx}$), networked nodes overspend energy dramatically. This is particularly pronounced in UWB-IR-based networks, where, due to the ratio between energy spending on receiving and transmitting (see Section 2.6.3), a node spends approximately 10 times more energy on receiving and forwarding a packet for another node than on transmitting its own

Algorithm 1 FLEX policy running on the CC, for the 3-node line multihop network topology.

Input: $s'_u(i), s'_v(i), s'_{cc}(i)$;
if $s'_u(i) > s'_v(i) \cdot (2 + c_{rx}/c_{tx})$ **then**
 $\widehat{s}_v(i) \leftarrow s'_v(i); \widehat{s}_u(i) \leftarrow s'_u(i) - s'_v(i) \cdot (1 + c_{rx}/c_{tx});$
else, $\{\widehat{s}_u(i), \widehat{s}_v(i)\} \leftarrow s'_u(i)/(2 + c_{rx}/c_{tx});$
end if
if $(s'_u(i)/c_{tx}) > (s'_{cc}(i)/c_{rx})$ **then**
 $R_T \leftarrow s'_{cc}(i)/c_{rx}; R_{curr} \leftarrow [\widehat{s}_u(i) + \widehat{s}_v(i)]/c_{tx};$
 if $R_{curr} > R_T$ **then**
 $\widehat{s}_u(i) \leftarrow \widehat{s}_u(i) \cdot R_T/R_{curr}; \widehat{s}_v(i) \leftarrow \widehat{s}_v(i) \cdot R_T/R_{curr};$
 end if
end if
Return: $r_u(i) \leftarrow \max[\widehat{s}_u(i)/c_{tx}, r_{min}]; r_v(i) \leftarrow \max[\widehat{s}_v(i)/c_{tx}, r_{min}];$

packet.

We evaluated flow control policies to which we refer to as FLEX, that are based on the DLEX node data rate assignment algorithm proposed in [57]. FLEX, running on the CC, assigns fair (*lexicographically maximal* [57]) data rates to the network nodes.¹⁷ Under FLEX, the prototypes independently determine their desired energy spending rates, $s'(i)$, using the EX or EP- K node energy allocation policies, and send them to the CC. The CC allocates data rates such that the total energy spending rates of the nodes do not exceed $s'(i)$, and the assigned data rates are fair. The FLEX policy for the 3-node line network topology (Fig. 2.14(a)) is shown in Alg. 1. The algorithm first computes lexicographically maximal data rates that the forwarding node u can support, then checks whether the CC can support these rates, and, if necessary, scales the rates proportionally. To maintain network connectivity, nodes communicate at a rate of at least

¹⁷The DLEX data rate allocation algorithm developed in [57] is implicitly tied to a particular, EP-based, node energy allocation policy. In FLEX, we combine the data rate allocation algorithm of [57] with different node policies.

r_{\min} .

We conducted extensive experiments with the FLEX policy using a variety of light inputs. Fig. 2.14(c), for example, demonstrates data rates assigned by FLEX, in combination with the EX and EP-1 policies, for a network with node energy harvesting rates shown in Fig. 2.14(b) (where errorbars represent variations in energy harvesting rates in different experiments). We provided the prototypes with light inputs corresponding to the light energy recorded over a day in L-2, compressed to 720s (12min) and scaled by 60.4x. The prototypes were equipped with OPVs. In 8 experiments, the variability in the total energy harvested was under 4.1% for the CC, and under 1% for nodes u and v . Despite providing nodes nearly identical light inputs, the harvesting rates, shown in Fig. 2.14(b), *differed by more than 1.8x*. This is due to different efficiencies of the OPVs integrated with different prototypes.

The FLEX policy, in combination with the EP-1, ensures energy neutrality. In combination with EX and with EP- K for $K \neq 1$, similarly to the DRC policy examined in the previous section, FLEX may *underspend or overspend* the energy of the nodes. For example, in the evaluation scenarios shown in Fig. 2.14, combined with EP-1, FLEX spends 96.4% of the energy harvested by node u ; combined with EX, it spends 125%. To achieve energy neutrality, FLEX needs to take into account energy spending on *control messages*, transmitted by the CC at a fixed rate r_c . For example, prior to calculating node data rates according to Alg. 1, nodes' $s'(i)$ values need to be reduced as $s'_{cc}(i) \leftarrow s'_{cc}(i) - r_c \cdot c_{tx}$, $s'_u(i) \leftarrow s'_u(i) - r_c \cdot [c_{rx} + c_{tx}]$, $s'_v(i) \leftarrow s'_v(i) - r_c \cdot c_{rx}$. We note that r_c values *directly affect nodes' data rates*, as can be seen, for example, in Fig. 2.14(c), which corresponds to energy harvesting rates shown in Fig. 2.14(a).¹⁸ In future work, we will develop flow control policies that jointly determine, and adapt to environmental energy dynamics, node energy allocations, control message rates, and data rates.

2.7.2 Collection Tree Selection

We evaluated collection tree adaptation policies in a ‘diamond’ network topology shown in Fig. 2.15(a), where nodes u and v send messages to the CC directly, while node w sends its

¹⁸Each data point in Fig. 2.14(d) corresponds to a complete 700s (11.6min) testbed experiment.

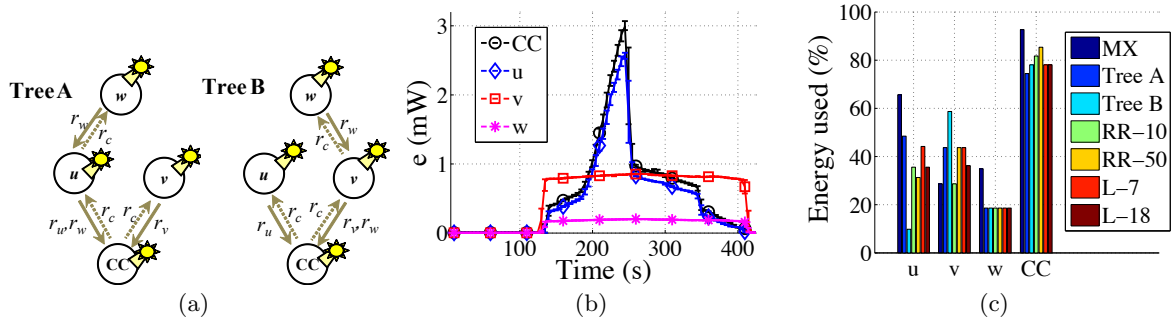


Figure 2.15: Topology adaptation policies for (a) a 4-node ‘diamond’ network: Using a day-long irradiance trace compressed to 400s and scaled by 20.0x from L-1, L-2, and the corresponding (b) energy harvesting rates from 8 experiment repetitions with errorbars, and (c) percentage of harvested energy used by the nodes.

Policy	Tree A	Tree B	RR-10	RR-50	L-7	L-18
\bar{r} , % of MX	82.4	95.1	92.6	97.8	88.2	92.8

Table 2.4: Data rates \bar{r} under different topology adaptation policies.

messages via a forwarder, u (tree A) or v (tree B). A collection tree is chosen by the CC based on one of the heuristics outlined below. Once a tree is selected, node data rates are assigned using the FLEX flow control policies, in combination with the EX policies. We compare the performance of these heuristics to the performance in networks where the collection trees are fixed, and to an EP-1-based *MX policy*, that calculates the best collection tree offline.

Round Robin (RR- T): Trees A and B alternate every T seconds. We evaluate the RR- T policy for $T = 10$ s (RR_{10}) and $T = 50$ s (RR_{50}).

Battery Level-based (L- B): The collection tree is changed, if the battery level of the forwarder in the current tree is B mC lower than the battery level of the other forwarder. We evaluate the L- B policy for $B = 7$ mC (L_7) and $B = 18$ mC (L_{18}). We note that tree selection based on node battery levels is commonly used in (non-energy-harvesting) sensor networks (e.g., [168]).

The evaluation results presented below correspond to the node energy harvesting rates shown in Fig. 2.15(b) (errorbars represent variations in harvesting rates in different experiments). We

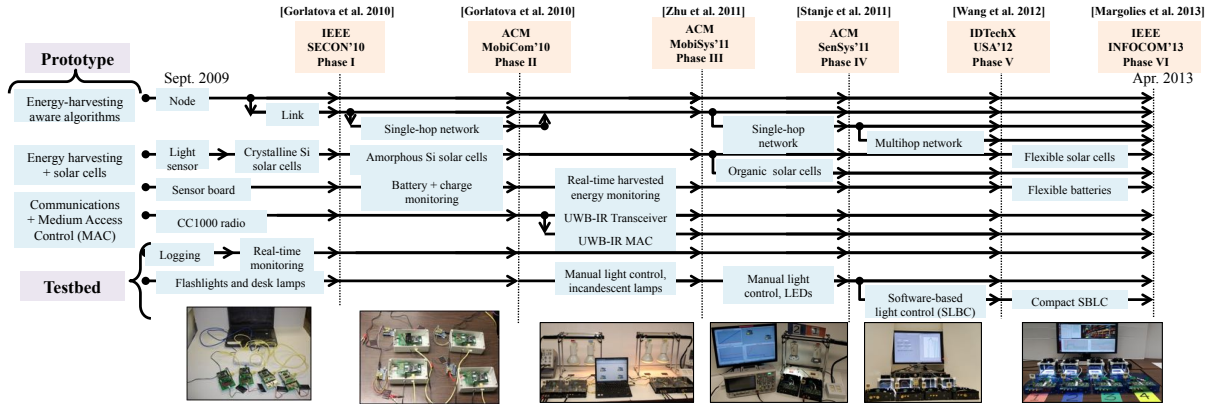


Figure 2.16: A timeline of the prototype and the testbed development stages.

provided the prototypes with light inputs corresponding to the light energy recorded over a day in L-1 and L-2 (nearby locations in the same office [71]), compressed to 400s (6.6min) and scaled by 20.0x. The prototypes were equipped with OPVs. In 8 experiments, the variability in total energy harvested was under 3% for all nodes.

Table 2.4 shows the average data rates obtained under different policies, as a percentage of data rates obtained under the MX policy. Note that using Trees A and B results in different data rates (due to the difference in the energy harvested by u and v). Fig. 2.15(c) shows the percentage of harvested energy used by the nodes.

All collection tree adaptation policies resulted in data rate improvements over the fixed Tree A. This emphasizes the need for topology adaptation policies for networks of energy harvesting nodes. The use of the L - B policies *did not result in data rate improvements* over the fixed Tree B. Due to FLEX and EX not taking battery levels into account, L - B only changes energy use patterns in nodes u and v (as can be seen in Fig. 2.15(c)). Developing and implementing policies for joint data rate determination and topology adaptation for EnHANTs is a subject for future work.

2.8 Lessons Learned

In this section, we first summarize the design and development process for the different prototype

components and the testbed. Then, we list several lessons that were learned throughout the process.¹⁹ We separate the discussion to lessons learned in the design and development of the prototypes and lessons learned in the design and development of the testbed.

2.8.1 Evolutionary System Development

The prototypes and the testbed were developed and integrated in a number of phases over a 4-year period, as shown schematically in Fig. 2.16.²⁰ We initially used off-the-shelf components; throughout the different phases, nearly all the off-the-shelf components were replaced with custom-designed ones. At the end of each phase, we integrated the phase's prototype and testbed components, and demonstrated a fully functional system. This approach necessitates continuous interaction between the designers and developers of different components and algorithms, and ensures that the integration challenges are addressed quickly. We summarize the functionality development steps below.

- **Energy harvesting** allows EnHANTs to self-power by obtaining energy from ambient sources. We have fabricated *flexible solar cells* that efficiently harvest indoor light, and integrated them with the prototypes. Initially, we designed the prototypes to sense, but not harvest, available environmental energy (Phase I). Next, we integrated rigid commercial solar cells and implemented real-time energy harvesting state monitoring (Phases II and III). Finally, we integrated the custom-designed organic solar cells (Phase V) and commercial flexible solar cells (Phase VI).
- **Ultra-Wideband Impulse-Radio (UWB-IR) wireless communications** spend significantly less energy than other low-power wireless technologies [44]. Early-phase prototypes communicated with each other via standard (non-UWB) commercial sensor network mote transceivers [4]. Prior to integration of the custom UWB-IR communication modules

¹⁹While this chapter focuses on the technical lessons learned, we also learned several organizational lessons, which are summarized in Appendix 6.

²⁰As described in Section 2.3, all experiments were conducted using the Phase V version of the prototype and testbed from Fig. 2.16.

in Phase III, we substantially modified the mote operating system (which did not support custom transceivers). The integration additionally required the implementation of a custom medium access control module, since the UWB-IR transceiver characteristics differ greatly from the properties of the conventional transceivers.

- **Energy harvesting-adaptive algorithms** were first designed and developed for simple single node scenarios, and were later implemented for network scenarios. Following the integration of the UWB-IR transceivers in Phase III, we re-implemented the algorithms to take the UWB-IR characteristics into account. In Phase VI, we introduced an adaptive multihop network.
- **Testbed functionality** first consisted of a data logger with a simple visualization interface, which we replaced with a custom-designed real-time monitoring and control system. We additionally developed several prototype light energy control systems, from relatively simple manual setups (Phases III and IV) to a software-based system that exposes the prototypes to real-world trace-based light energy conditions (Phase V).

At the end of each phase, a fully functional prototype testbed was demonstrated, as indicated in Fig. 2.16.

2.8.2 Designing and Prototyping EnHANTs: Bridging the Theoretical and Experimental Gap

While designing, developing, and evaluating energy harvesting adaptive networking algorithms, we aimed to bridge theory and experiments. However, the majority of the practical problems we ran into are considered out-of-the-scope of theoretical studies. Hence, we list a few *gaps* between our experience and the common theoretical assumptions.

- Exact energy spending mechanisms and detailed energy storage state tracking are very difficult to implement, and their implementation places substantial loads on the prototype (i.e., battery sampling overhead). This observation is in contrast to the assumptions made

by many theoretical energy harvesting adaptive algorithms which commonly rely on the availability of such mechanisms [51, 52, 168].

- In many theoretical studies on energy-harvesting adaptive networking, the hardware parameters are taken for granted and *are not part of the solution space*. Therefore, many of the parameters of the energy harvesting adaptive algorithms that we evaluated corresponded to open theoretical problems. Specifically, to the best of our knowledge, *there are no guidelines* on issues such as determining the optimal battery size for energy harvesting nodes, optimal energy storage quantization level and update frequency, and network adaptation recalculation interval. Additionally, we found that accounting for a minimum link rate r_{min} (Section 2.6.5) and a network control traffic rate r_c (Section 2.7.2) complicated maintaining energy neutrality in the network's nodes. This is a phenomenon which is often ignored.
- Theoretical studies on MAC for UWB-IR (i.e., [49]) typically assume the existence of complex physical layer implementations and therefore cannot be evaluated in a testbed. On the other hand, even basic theoretical MAC layer algorithms (e.g., Aloha and CSMA) are difficult to implement using experimental hardware. The lesson learned is that MAC layer evaluation using experimental hardware can provide grounding assumptions for theoretical studies.
- Policies optimized for an individual layer of the network (e.g., node, link) may not perform well when combined together and can be significantly affected by the choice of physical components. For example, we observed that using different types of solar cells results in different dynamics in energy harvesting (see Fig. 2.7(c)). Similarly, different transceivers have *very different levels of physical layer reliability* (i.e., bit error rates). Hence, when we switched from CC1000 radios to the UWB-IR transceivers in Phase III (see Fig. 2.16), we had to redesign the higher layer algorithms.

2.8.3 Testbed Design and Development

2.8.3.1 Debugging Interfaces

In system design with many components, debugging interfaces are critical. Observing critical system parameters while not interfering can be a significant challenge. In our testbed, we have many components which ideally require separate debug interfaces. In particular,

- The UWB-IR Communication Module (see Fig. 2.11) contains chipset debugging signals which are observed through added ports using an oscilloscope. Additionally, the module contains jumper connections and potentiometers to modify physical layer parameters.
- The prototypes are placed on MIB600 programming boards to enable an Ethernet control channel. However, sending data over the control channel incurs a computational load on the CM and hence, it must be used sparingly.
- The EHM was designed and verified separately, and incorporated with the rest of the prototype without additional debug interfaces. Namely, in the testbed, we could observe B and s values reported by the EHM, but had no means of verifying them. That made designing and verifying energy harvesting adaptive algorithms, which rely on EHM-reported values and control EHM operations, very difficult. Other similar systems (e.g., [5]) should consider having debug interfaces added for energy monitoring.

As a consequence of debugging interfaces, the prototype is much larger than envisioned. Although reducing the size is not one of our current goals, it is certainly a challenge to do it while maintaining the debugging capabilities.

2.8.3.2 Facilitating Repeatable Experiments in an Energy Harvesting Testbed

To the best of our knowledge, the EnHANTs testbed is the only energy harvesting testbed that is able to support repeatable experiments. This capability requires generation of physical phenomena and is necessary, since energy generated by other means (i.e., the environment) has uncontrollable order-of-magnitude variations. Below, we outline some of the lessons learned from implementing this functionality.

- The software-based light control system exposes individual prototypes to repeatable light energy conditions (see Section 2.3.2). Providing *controllable and repeatable* trace-based light energy conditions proved to be very challenging process, involving a painstaking calibration which utilized NIST-traceable photodiodes. Moreover, the energy spending of the prototype, which is performed via activation of a load resistor, also had to be carefully tuned to account for variations in battery voltage and the internal capacitance of the components.
- Lights are *point energy sources* and thus small changes in photovoltaic cell position changes the energy harvesting rates dramatically. Hence, we learned the importance of precise and reproducible photovoltaic cell placement. Therefore, the photovoltaics were carefully placed inside custom-designed dark box enclosures, shown in Fig. 2.5.
- As light and heat are inherently related when placing lamps close to the nodes, the solar cells are affected by the heat. However, it is impractical to place lamps far from the nodes. We observed many effects related to heat in solar cells; our experiments demonstrated a loss of up to 7.6% of the harvested energy as solar cells were heating up. In one of our early designs, the intense heat generated by incandescent lamps affected not only the performance of solar cells, but also *the clock frequency drift of the UWB-IR Communication Module*. Hence, we learned the importance of minimizing heat generation and utilized cool white LEDs which were positioned external to the prototype.

2.9 Ongoing and Future Work

In our ongoing work we have been focusing on developing hardware, software, and algorithms that will support the realization of the long-term EnHANTs vision. Namely, we have been improving the design of the UWB-IR communication module, integrating flexible batteries, and implementing an adaptive multihop network (e.g., [112]). We now list some of the directions for future work.

- **Testbed and Control and Monitoring System:** From the prototypes design and

testbed experiments, we learned that energy harvesting forces a complete rethinking of the protocol stack as well as the node architecture. However, while our small testbed provides a proof of concept, it does not allow for large scale networks experimentation. Therefore, in our future work we will enhance the existing testbed. Namely, we will design and deploy a large testbed composed of the next generations of prototypes. The testbed will allow for evaluating experiments with un-compressed and un-scaled light traces as well as for tracking the detailed energy consumption from the various components. Furthermore, we will also have an option to evaluate nodes in *live environments*, where they will harvest energy from ambient light sources instead of the software controlled lights.

The prototypes will be modified for streamlined deployment in a large testbed. Non-tag functionality will be moved from the prototype CM to a backbone network with a control and monitoring system. The CM in the current prototype deals with both tag-essential functionalities (e.g., MAC, routing, energy management) and non-tag-essential tasks such as logging and event transmission for the purpose of monitoring the experiments. The monitoring tasks unnecessarily stress the CM in terms of power, memory, and performance requirements. The separation will enable further optimizations for both the prototypes and the backbone network control and monitoring system.

- **Microcontroller:** As we are pushing the limits of performance and energy efficiency via a range of custom hardware and software, the legacy MICA2 hardware platform (that has been used for the CM) has been found to be a bottleneck for further research explorations. Therefore, we will integrate a KMote [93] as a dedicated EnHANTs CM board for the tag-essential tasks. The KMote uses the 16b Texas Instruments MSP430 [154] which is a state-of-the-art low-power microcontroller. It offers substantial power savings and performance improvements over the 8b Atmel AVR [19] used in MICA2 motes. Through preliminary testing, we have already verified the compatibility of the KMote with the EHM and UWB-M. To allow for a smooth transition of our existing software, drivers, and algorithms, and to enable relatively quick deployment, we plan to initially use the same operating system as in the current testbed TinyOS version 2.1.2 [156], which also supports

the MSP430 platform. In parallel to the development process, we will evaluate other low power operating systems like Contiki [43] or Lorien [106] as potential upgrades.

- **Application Optimization:** The design of the lower layers is usually driven by application requirements. Therefore, we plan to develop tracking and monitoring applications and to evaluate their performance along with various combinations of lower layer algorithms (e.g., harvesting-aware MAC and forwarding) and hardware configurations (e.g., UWB transceiver settings, computational architectures, and energy storage components). We will, for example, develop and evaluate the algorithms for specific applications including detecting misplaced objects, searching for objects, or obtaining and maintaining an inventory list of objects.
- **Alternate Energy Sources:** The light control system can be used to provide the EHM with traces that were collected from other energy sources. In particular, since kinetic (movement) energy availability is different from light energy, the developed algorithms should be different [64]. The ability to provide the nodes energy traces and emulate a network of kinetic harvesting nodes without the need to fabricate (and move around) kinetic harvesters will be an important step towards the use of this technology for networking applications.
- **Multi-tag Synchronization in UWB-IR:** Maintaining synchronization in impulse-radio requires novel UWB-IR receiver architectural approaches. When nodes are unsynchronized, the high cost of non duty-cycled reception while searching for the packet start results in significant energy waste. Synchronization can then be maintained through the use of a light-weight overhead connection. This consumes some energy, however, and when to keep or drop synchronization is an open area of research to be investigated. One possible approach is to use immediate hardware initiated acknowledgments and leverage the energy savings of the existing synchronization. We aim to substantially reduce synchronization energy costs through the development of a cross-layer protocol, optimized to application requirements.

- **Energy Harvesting Aware MAC:** Similarly to algorithms in the higher layers (e.g., flow control), the MAC protocols will have to adapt to harvesting and battery states of the node and neighbors. For example, in some cases, it may be beneficial to create a high power mode and spend more energy than what is typically spent by a tag (e.g., when the EHM reports the battery is fully charged and the tag is harvesting energy). Other examples include energy harvesting adaptive sleep scheduling. However, implementing sleep scheduling with UWB-IR is non-trivial (due to bit synchronization issues discussed above). Due to the paradigm shift in the receive and transmit cost, we will consider strategies such as receiver initiated wake up [51] that shift the energy demand of sending a packet from the receiver to the sender.
- **Form Factor:** We will design and integrate with the prototypes hardware components tailored for the future EnHANT form factor. The current prototypes (Fig. 1.5) are much larger than the envisioned EnHANT (Fig. 1.2). The current prototypes include a number of debug interfaces and test points to aid development. After the completion of the prototype development, the core electronic components, including the microcontroller, UWB-IR transceiver, and energy management circuits, will be combined into a single custom Integrated Circuit (IC). This IC will then be mounted on a flexible Printed Circuit Board (PCB) that includes an integrated antenna, thin-film solar cell, and thin-film battery.

A physical mock-up of an envisioned EnHANT is shown in Fig. 1.2(b). This mock-up consists of a thin-film battery ($2.5 \times 5.8 \text{ cm}^2$, 2.2mAh, 31.6J) [83] and thin-film solar cell ($3.65 \times 6.4 \text{ cm}^2$) [127] attached to a PEN plastic substrate an anisotropic conductive paste and contact points that allow the solar cell and battery to be used in the existing testbed. This simple mock-up demonstrates that it is possible to build a functional tag with the desired form factor. Based on an efficiency of the solar cell at 1% and an irradiance level of $1 \text{ J/cm}^2/\text{day}$ [68,71], this tag could harvest 0.233 J/day and could maintain a continuous data rate of 2.7kbps using the UWB-IR transceiver. The mock-up does not include an antenna, but previous research has demonstrated designs for UWB antennas that are less than $3 \times 3 \text{ cm}^2$ [35].

2.10 Conclusions

This chapter describes the design and development of the Energy Harvesting Active Networked Tag (EnHANT) prototypes and testbed, as well as experiments with energy harvesting adaptive policies. To the best of our knowledge, the prototypes and the testbed are *the first of their kind*. The design, development, and evaluation of the EnHANTs prototypes, testbed, and algorithms brings about various *lessons learned* that often cannot be modeled. These lessons learned include, for example, the difficulty in implementing precise energy spending and tracking mechanisms, that small changes in the photovoltaic type or position can significantly change the energy harvesting rates, and that policies optimized for an individual layer of the network (e.g., node, link) may not perform well when combined together and can be significantly affected by the choice of physical components. We note that these contributions are applicable to various networks of energy harvesting devices, not just the EnHANTs.

This page intentionally left blank.

Chapter 3

Panda: Neighbor Discovery on a Power Harvesting Budget

Object tracking applications are gaining popularity and will soon utilize Energy Harvesting (EH) low-power nodes that will consume power mostly for Neighbor Discovery (ND) (i.e., *identifying* nodes within communication range). Although ND protocols were developed for sensor networks, the *challenges posed by emerging EH low-power transceivers were not addressed*. Therefore, we *design an ND protocol tailored for the characteristics of a representative EH prototype*: the TI eZ430-RF2500-SEH.

We present a generalized model of ND accounting for unique prototype characteristics (i.e., energy costs for transmission/reception, and transceiver state switching times/costs). Then, we present the Power Aware Neighbor Discovery Asynchronously (Panda) protocol in which nodes transition between the sleep, receive, and transmit states. We analyze Panda and optimize its parameters subject to a homogenous power budget. We also present Panda-D, designed for *non-homogeneous* EH nodes. We perform extensive testbed evaluations using the prototypes and study various design tradeoffs. We demonstrate a small difference (less than 2%) between experimental and analytical results, thereby confirming the modeling assumptions. Moreover, we show that Panda improves the ND rate by up to 3x compared to related protocols. Finally, we show that Panda-D operates well under non-homogeneous power harvesting.

This work is also pending submission to a conference [110] and has a provisional patent [111]. Additionally, numerous design contributions were made by Ph.D. Candidate T. Chen and by Dr. G. Grebla.

3.1 Introduction

Object tracking and monitoring applications are gaining popularity within the realm of the Internet-of-Things [20]. Emerging low-power wireless nodes that can be attached to physical objects are enablers for such applications. Often, these nodes are meant to interact with a reader, but architectures are emerging that handle scenarios where no reader may be present, or where the number of nodes overwhelms the readers' availability. These scenarios can be supported by Energy Harvesting (EH) tags (e.g., [61, 158] and references therein) that are able to communicate peer-to-peer and are powered by an ambient energy source (e.g., light).

Such EH nodes will enable tracking applications in healthcare, smart buildings, assisted living, manufacturing, supply chain management, and intelligent transportation as discussed in [103, 161]. An example application, illustrated in Fig. 1.6(a), is a large warehouse that contains many inventory items, each of which is equipped with an EH node. Each node has an ID that corresponds to the physical object (item). The nodes utilize a Neighbor Discovery (ND) protocol to *identify* neighbors which are within communication range, and therefore, the system can collect information about the objects' whereabouts. A simple application is identifying misplaced objects: often when an item is misplaced (e.g., in a furniture warehouse, a box of table parts is moved to an area with boxes of bed parts), its ID is significantly different from the IDs of its neighbors. In such a case, the misplaced node can, for instance, flash a low-power LED to indicate that it is lost.

In this chapter, *we develop an ND protocol for Commercial Off-The-Shelf (COTS) EH nodes, based on the TI eZ430-RF2500-SEH* [5] (shown in Fig. 1.6(b)). The nodes harvest ambient light to supply energy to a low-power microcontroller and transceiver. To maintain *perpetual* tracking of the (potentially) mobile objects, ND must be run continuously with the node operating in an *ultra-low-power mode* that consumes power at the rate of power harvested [69]. Our objective

is to maximize the rate that nodes *discover* their neighbors, given a constrained power budget at each EH node.

ND has always been an important part of many network scenarios [131,148]. Yet, to consume power at the rate of power harvested, EH nodes require *extremely limited power budgets*: we show that, even with optimized power spending, the duty cycles are between 0.1-0.6%. Therefore, numerous assumptions from related works (e.g., [24,149]) no longer hold, including that switching times (between the sleep, receive, and transmit states) draw negligible power and that the power costs to send and receive are identical (see Section 3.2 for details). Furthermore, in the envisioned applications, the node’s main task is to perform ND, and thus, the power consumed by ND is the dominant component of the power budget.

Hence, we design, analyze, and experiment with **Panda— Power Aware Neighbor Discovery Asynchronously**,¹ an ND protocol that *maximizes the average discovery rate under a given power budget*. The main contributions of this chapter are:

(C1) Radio Characterization: We model a generic ultra-low-power EH node that captures the capabilities of our prototype (Fig. 1.6(b)). We also study, for the first time, important properties of the radio in the context of ND. We show that characteristics such as the power consumption and the time to transition between the different states (e.g., sleep to listen) are crucial to incorporate into the design of ND protocols for EH nodes.

(C2) Panda Protocol: We develop the Panda protocol in which an EH node discovers its neighbors by transitioning between the sleep, receive, and transmit states at rates that satisfy a power budget. Furthermore, we present Panda-Dynamic (Panda-D), which extends Panda’s applicability to non-homogeneous power harvesting and multihop topologies.

(C3) Protocol Optimization: Using techniques from renewal theory, we derive closed form expressions for the discovery rate and the power consumption. We develop the Panda Configuration Algorithm (PCA) to determine the node’s duration in each state (sleep, receive, transmit), such that the discovery rate is maximized, while meeting the power budget. The solution obtained by the PCA is numerically shown to be between 94–99.9% of the optimal for all scenarios

¹The protocol name, Panda, relates to the animal as both EH nodes and Pandas spend the majority of their time sleeping to conserve energy.

considered.

(C4) Experimental Evaluation: Using TI eZ430-RF2500-SEH EH nodes [5], we show that the real-life discovery rates are within 2% of the analytically predicted values, demonstrating the practicality of our model. Moreover, we show that Panda’s *experimental* discovery rate is up to 3 times higher than the discovery rates from simulations of two of the previously best known low-power ND protocols [24, 115]. Furthermore, we demonstrate that Panda-D adjusts the rate of ND for scenarios with non-homogenous power harvesting and multihop topologies.

The rest of the paper is organized as follows. In Section 3.2 we discuss related work. In Section 3.3 we present the system model. In Sections 3.4 and 3.5, we present and optimize Panda, respectively. In Section 3.6, we present the Panda-D protocol. In Section 3.7, we evaluate Panda experimentally. We conclude in Section 3.8.

3.2 Related Work

ND for low-power wireless networks is a well studied problem (see [20, 131, 148] for a summary). The protocols can be categorized into deterministic (e.g., [24, 53, 88, 129, 149]) and probabilistic (e.g., [115, 173]). Deterministic protocols focus on guaranteeing an upper bound on *discovery latency*, while the choice of parameters (e.g., prime numbers) is often limited. On the other hand, the most well-known probabilistic protocol [115], has a better average ND rate, but suffers from an unbounded discovery latency. Our probabilistic protocol, Panda, is fundamentally different: other protocols (i) are constrained by a duty cycle, instead of a power budget, (ii) do not account for channel collisions (e.g., when two nodes transmit at the same time), (iii) rely on each node maintaining synchronized time slots,² or (iv) do not consider practical hardware energy consumption costs (i.e., the power consumed by the radio to transition between different states).

To the best of our knowledge, *Panda is the first ND protocol for EH nodes and the first attempt to maximize the discovery rate, given a power budget.* As such, Panda will operate

²It was shown in [53] how the aligned time slot assumption can be relaxed. Yet, practical considerations such as selecting the slot duration and avoiding collisions are not described.

with duty cycles between 0.1-0.6%, which is an order of magnitude lower than those typically considered in prior works [148].

In our experiments, we use hardware from [5]. There are also numerous other hardware options for EH nodes [2, 61, 151], computational RFIDs [177], and mm³-scale wireless devices [3]. Additionally, there are other radio features that achieve low energy consumption. For example, preamble-sampling and wake up radios were investigated in [55] and [50], respectively, for WSNs. However, the added power consumption of these features makes them impractical for the EH nodes we study. Furthermore, numerous options for low-power wireless communication exist (e.g., Bluetooth Low Energy [9]). However, [5] is one of the increasingly popular low-power COTS EH nodes which seamlessly supports wireless protocol development.

3.3 System Model

In this section, we describe our prototypes, based on which, we introduce the notation and the system model.

3.3.1 Prototype Description

The prototype is shown in Fig. 1.6(b) and is based on the commercially available TI eZ430-RF2500-SEH [5]. We made some modifications to the hardware, which are summarized in Appendix 3.9.1. We now describe the prototype's components:

Energy Harvesting Power Source: The prototype harvests light from a Sanyo AM 1815 amorphous solar cell [137]. The solar cell is set to a fixed harvesting voltage of 1.02V (no power point tracking techniques are used). To measure the power harvested, we place a multimeter in series with the solar cell.

Energy Storage: The energy harvested by the solar cells is stored in a capacitor and the voltage is denoted by V_{cap} . The voltage is regulated to 3.5V to power the node. We modified the board design to enable experimentation with varying capacitor sizes. Unless stated otherwise, we use a 30mF capacitor. To ensure stable operation of the microcontroller, the capacitor is prevented from reaching a low voltage level by imposing a software cutoff when $V_{\text{cap}} = 3.6\text{V}$. If

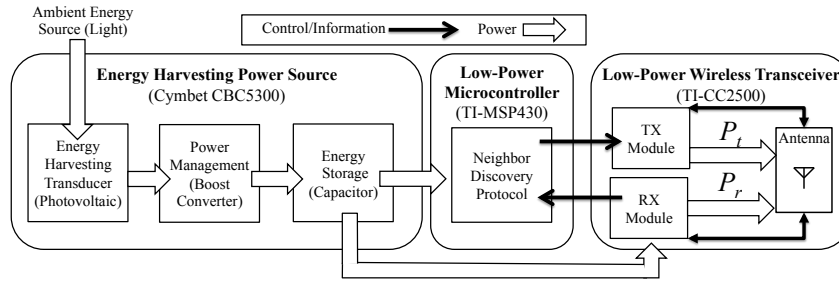


Figure 3.1: A hardware model of the EH node (based on the TI eZ430-RF2500-SEH [5]).

a node reaches such a low voltage, it enters a low-power sleep state for 10s or until the voltage level reaches a safe operating point.

Low-Power Microcontroller: A TI-MSP430 microcontroller [154] is used to provide computational capabilities. These include (i) sampling the capacitor voltage using an analog to digital converter (ADC), (ii) operating a low-power 12kHz clock with an idle power draw of 1.6 μ W to instruct the node to enter and exit an ultra-low-power sleep state, and (iii) receiving and sending messages to the radio layer.

Low-Power Transceiver: The prototype utilizes a CC2500 wireless transceiver (a 2.4GHz transceiver designed to provide low-power wireless communication) [8] to send and receive messages. The transceiver operates at 250kbps and consumes 64.85mW while in receive state. The transmission power can be set in software and we utilize levels between -16 and 1dBm, with a resulting power consumption between 53.25 and 86.82mW. At these levels, nodes within the same room typically have little or no packet loss.

3.3.2 System Model

The model is based on the prototype and is shown pictorially in Fig. 3.1. Yet, it is generalizable to a class of other prototypes (e.g., [61]). A summary of the nomenclature from this point forward appears in Table 3.1.

A node can be in one out of three states, denoted by the set $\mathcal{S} = \{s, r, t\}$ for sleep (s), receive³ (r), and transmit (t). A node in state $i \in \mathcal{S}$ consumes power of P_i . Since the power

³We refer to the *receive* and the *listen* states synonymously as the power consumption of the prototype in both

Symbol	Description
V_{cap}	The voltage of the capacitor (V)
N	Number of nodes
P_b	Average power spending budget (mW)
P_t	Transmitting power consumption (mW)
P_r	Listening/Receiving power consumption (mW)
C_{ij}	Energy cost to transition from state i to j (μJ).
M	Discovery-packet duration (ms)
λ	Rate of exponential distribution (ms^{-1})
l	The duration of the listening period (ms)
ρ	Expected renewal duration (ms)
Y	Denotes role of node in the renewal
$\eta()$	Expected energy spending (μJ)
$\Phi()$	Expected power spending in a renewal (mW)
χ	Expected duration of idle listening (ms)
U	Discovery rate (second^{-1})

Table 3.1: Nomenclature

consumption in sleep state is negligible, we assume $P_s = 0$ throughout the chapter and remark that all results can be easily applied for $P_s > 0$, as described in Appendix 3.9.2. For the power budgets we consider, the energy consumed by the radio to transition between different states is non-negligible. Hence, we denote by C_{ij} the energy (μJ) consumed to switch from state i to state j ($i, j \in \mathcal{S}$).

Unfortunately, the prototype does not have explicit power awareness (unlike, e.g., [61]). Therefore, we impose a power budget, P_b (mW) on each node. The power budget is set such that *energy neutrality* is achieved: nodes consume power (on average) at the power harvesting rate [159]. Hence, for an EH node harvesting more power (e.g., brighter light source), the power budget P_b is higher.

We denote by N the number of nodes in the network and present two important definitions:

states is similar.

Definition 1. The **discovery message** is a broadcast packet containing the ID of the transmitter.⁴ A **discovery** occurs when a node receives a discovery message from a neighbor. Multiple discoveries can occur per discovery message transmission.

Definition 2. The **discovery rate**, denoted by U , is the expected number of discoveries in the network per second.

The objective of the ND protocol is to *maximize the discovery rate, subject to a given power budget*. This is in contrast to other works which seek to minimize the worst case discovery latency [24, 149], subject to a duty cycle. As such, in Section 3.7, we also consider the discovery latency, or time in between discoveries, as a secondary performance metric.

3.4 Panda Protocol

In this section we describe and analyze Panda, an asynchronous ND protocol, which operates under a power budget.

3.4.1 Protocol Description

Fig. 3.2 depicts the state transition diagram for the Panda protocol, from sleep to listen to transmit and then back to sleep. To ensure perpetual operation under the power budget P_b (mW), nodes initialize in a low-power sleep state to conserve energy. To maximize the discovery rate, Panda follows a probabilistic approach in which nodes sleep for an exponential duration with rate λ (ms^{-1}). The probabilistic sleep duration prevents unwanted synchronization among subsets of nodes.

Following sleep, nodes awaken and listen to the channel for discovery messages from their neighbors for a fixed duration of l (ms). If a message is received, the node remains in the listen state until it completes reception of this message. If no transmission is heard while in the listen state, the node transmits its discovery message of fixed duration M (ms).⁵

⁴In practice, the discovery message may include information on already discovered neighbors, thus enabling

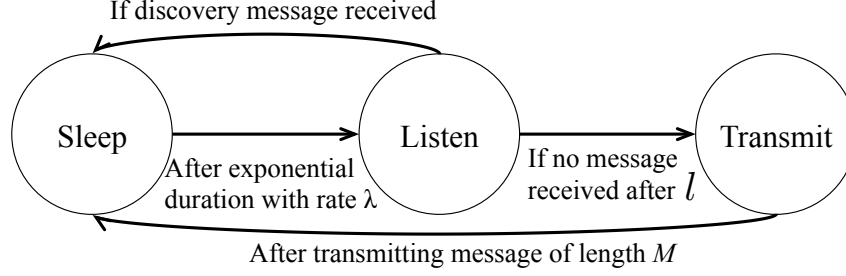


Figure 3.2: Panda protocol outline: EH nodes transition between radio states (sleep, listen, and transmit) to maintain within a power budget.

Note that in Panda, similar to CSMA, nodes always listen before they transmit, and therefore, there are no packet collisions between two nodes in wireless communication range of one another. Additionally, after a message is transmitted, the node returns to the sleep state. Hence, there is no acknowledgement of the discovery. This is because coordinating acknowledgement messages among multiple potential receivers can be costly, requiring additional listening by the transmitter and possibly collision resolution.

3.4.2 Analysis

While Panda can operate in general scenarios, for analytical tractability, we assume the following:

- (A1) All nodes are homogenous, namely, have the same power budget P_b and the same hardware.
- (A2) Every pair of nodes can exchange packets (clique topology) with no packet errors due to noise.
- (A3) The number of nodes, N , is known a priori.

These assumptions are applicable to some systems and envisioned applications. For example, when tracking boxes in a room (Fig.1.6(a)), these assumptions are close to reality as nodes in close proximity harvest similar amounts of energy, have few packets lost, and the number of nodes can be estimated a priori. However, for scenarios in which these assumptions do not hold,

indirect discoveries. However, we do not consider these indirect discoveries.

⁵The discovery message duration, M , is fixed, stemming from the fixed size of the node ID contained in the message.

in Section 3.6, we present Panda-Dynamic which is based on relaxed assumptions and discuss the implications.

We note that, as this is the first attempt to develop an ND protocol explicitly for EH nodes, it is natural to consider the homogeneity assumption (A1). Additionally, several other works also make assumptions similar to (A2) and (A3) (e.g., [24, 149] only consider ND for a link, $N = 2$, with no collisions or packet loss).

Using these assumptions, we now use techniques from renewal theory [36] to analyze Panda for a network of N nodes. The renewal process is shown pictorially in Fig. 3.3. The renewal initiates with all nodes in the sleep state and ends after one node completes its transmission, whether the message is heard or not. The sleep duration for each node follows a memoryless exponential distribution. Therefore, for all analytical purposes, all nodes effectively initiate their sleep state at the start of the renewal.

In each renewal, the *first* node to wake up begins its listen state, and after a duration l , it transmits its discovery message. This is exemplified by node 6 in Fig. 3.3; we denote by \mathcal{N}_t the set containing a single transmitting node in a renewal.

Nodes that are in the receive state (r) when a message transmission begins, will stay in this state until the transmission is completed and then switch to the sleep state (s). We denote by \mathcal{N}_r the set of such nodes and $|\mathcal{N}_r|$ the size of the set, exemplified by nodes 2-4 in Fig. 3.3. The expected idle listening time of a node in \mathcal{N}_r is denoted by χ . Fig. 3.3 shows examples of idle listening durations for nodes 2-4, denoted as χ_i . Any node which wakes up in the middle of the message transmission immediately senses the busy channel and returns to the sleep state. An example is node 5 in Fig. 3.3.

When the transmission is completed, all nodes are in sleep state and the renewal restarts. The average renewal duration is the time it takes for the first node to wake up (occurring with rate $N\lambda$), listen for a duration l , and transmit a message for a duration of M . Hence, the expected renewal duration ρ is:

$$\rho = 1/(\lambda N) + l + M. \quad (3.1)$$

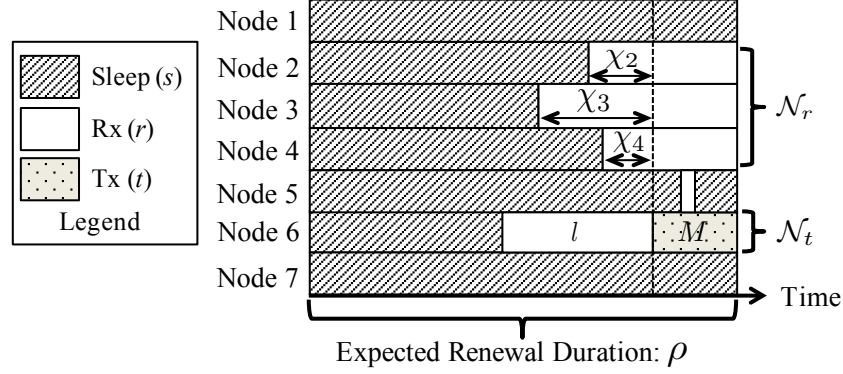


Figure 3.3: Renewal process representing one renewal cycle for $N = 7$ nodes: all nodes begin in a memoryless sleep state and the renewal restarts after the first node to wake up completes its transmission.

3.4.3 Discovery Rate

Recall that the objective of Panda is to maximize the discovery rate, U (see Def. 2). Considering U as the reward function and applying the elementary renewal theorem for renewal-reward processes [36], we obtain:

$$U := \lim_{t \rightarrow \infty} \frac{u(t)}{t} = \frac{\mathbb{E}[|\mathcal{N}_r|]}{\rho}, \quad (3.2)$$

where $u(t)$ represents the number of discoveries (as defined by Def. 1) by time t and ρ is computed by (3.1).

There are $N - 1$ nodes who are not the transmitter in the renewal, each of which is equally and independently likely to discover the transmitter. A discovery occurs, if the node wakes up from sleep in a period of time l preceding the transmission, an event with probability $1 - e^{-\lambda l}$. Hence,

$$\mathbb{E}[|\mathcal{N}_r|] = (N - 1)(1 - e^{-\lambda l}). \quad (3.3)$$

3.4.4 Energy Consumption

Since all nodes are homogenous (A1), we let n denote an arbitrary node and define a random variable Y that indicates the set $(\mathcal{N}_t, \mathcal{N}_r)$ in which the node resides in the renewal:

$$Y = \begin{cases} 0, & n \in \mathcal{N}_t \\ 1, & n \in \mathcal{N}_r \\ 2, & \text{otherwise.} \end{cases} \quad (3.4)$$

We let $\eta(y)$ represent the expected amount of energy (μJ) consumed by a node in a renewal in which $Y = y$. Thus,

$$\eta(0) = P_r l + P_t M + C_{sr} + C_{ts}, \quad (3.5)$$

$$\eta(1) = P_r(\chi + M) + C_{sr} + C_{rs}. \quad (3.6)$$

Eq. (3.5) defines the energy consumption of the transmitting node, which consumes energy to wake up from sleep, listen for a period of l , transmit a message, and then return to sleep. For a receiving node, the expected energy consumption is defined in (3.6) and consists of idle listening before the message transmission (with χ denoting the expected duration of idle listening, shown in Fig. 3.3). The derivation of χ can be found in Appendix 3.9.3. Then, the node listens for the duration of the message M .

Throughout this chapter, we assume that nodes which sleep for the entire renewal (e.g., nodes 1 and 7 in Fig. 3.3), and those which wake up briefly and sense a busy channel (e.g., node 5), do not consume power, and thus $\eta(2) = 0$. In Appendix 3.9.2, we justify this assumption and show how it can be relaxed.

The computation of $\Pr(Y = y)$ for $y = 0, 1$ is as follows. By definition of the renewal, there will be exactly one transmitter in a renewal and due to assumption (A1), $\Pr(Y = 0) = 1/N$. All the remaining $N - 1$ nodes successfully receive the message, if they start listening in a period of length l preceding the transmission. Hence, since the sleep duration is exponentially distributed, $\Pr(Y = 1) = (1 - e^{-\lambda l})(N - 1)/N$.

Define $\Phi(y) = \Pr(Y = y)\eta(y)/\rho$ and note its units are (mW); we will often refer to $\Phi(0)$ as the *probing power* while $\Phi(1)$ is referred to as the *discovery power*. As described above, $\eta(2) = 0$,

and thus $\Phi(2) = 0$. The expected power consumed by a node in a renewal must meet the power budget, i.e., $\Phi(0) + \Phi(1) \leq P_b$.

3.5 Optimization of Panda

Clearly, the choice of the sleep rate (λ) and the listen duration (l) determines the power consumption of the node as well as the discovery rate U . First, we demonstrate that an analytical solution is difficult to obtain. Next, we describe the Panda Configuration Algorithm (PCA) which obtains the *configuration parameters* (λ, l) for Panda. Finally, we demonstrate that the PCA obtains a nearly-optimal discovery rate.

3.5.1 Problem Formulation and Preliminaries

Finding (λ^*, l^*) that maximizes U is formulated as follows:

$$\max_{\lambda, l} \quad U = \frac{(N-1)(1 - e^{-\lambda l})}{\rho} \quad (3.7)$$

$$\text{s.t.} \quad \Phi(0) + \Phi(1) \leq P_b, \quad (3.8)$$

where (3.7) is derived using (3.2) and (3.3). Recall that ρ is computed from (3.1) and $\Phi(y)$ is computed using the results from Section 3.4.4. The problem as formulated above is non-convex and non-linear, and is thereby challenging to solve.

In the following subsections, we will attempt to find nearly-optimal Panda configuration parameters (λ, l) . We now provide several observations on the specific structure of the problem which are used throughout this section. First, we find the following Taylor-series approximation useful:

$$e^{-x} \geq 1 - x \text{ for } x \geq 0, \text{ and } e^{-x} \approx 1 - x \text{ for } x \approx 0. \quad (3.9)$$

We substitute x with λl in (3.9),⁶

$$U \leq \frac{(N-1)\lambda l}{\rho} := \bar{U}. \quad (3.10)$$

⁶Limited power budgets cause EH nodes to be in the sleep state much longer than in the listen state. Thus, $\lambda l \approx 0$ and (3.9) is a good approximation.

Panda Configuration Algorithm (PCA)

```

1: for  $K = [0, \epsilon, 2\epsilon, \dots, \lfloor \frac{\rho_{\max}}{\epsilon} \rfloor \epsilon]$  do
2:   Find  $(\lambda, l)$  that maximize (3.10) subject to (3.12)
3:   if  $\lambda, l$  satisfy (3.8) then
4:     Compute the discovery rate  $U$ 
5:   end if
6: end for
7: return  $(\lambda, l)$  that maximize  $U$ , denoted as  $\lambda_A, l_A$ , and  $U_A$ .

```

3.5.2 Panda Configuration Algorithm (PCA)

In this subsection, we present the Panda Configuration Algorithm (PCA) that returns a configuration of λ and l satisfying (3.8). To find a configuration with the highest discovery rate, the PCA utilizes a relaxed problem formulation as follows. An upper bound on the discovery power, $\bar{\Phi}(1)$, is computed by using (3.9) to obtain $(1 - e^{-\lambda l}) \leq \lambda l$, which leads to,

$$\Phi(1) \leq \bar{\Phi}(1) := \frac{N-1}{N\rho} \lambda l (P_r(\chi + M) + C_{sr} + C_{rs}). \quad (3.11)$$

The relaxed power budget constraint is then,

$$\text{s.t. } \Phi(0) + \bar{\Phi}(1) \leq P_b. \quad (3.12)$$

The PCA analytically computes the values of (λ, l) that maximize \bar{U} by solving for λ in terms of l in (3.11), and then finding the critical points where $d\bar{U}/dl = 0$. For computation tractability, the PCA replaces χ with a constant K in $\bar{\Phi}(1)$. The PCA uses the fact that, in practice, a node's sleep time is upper bounded, introducing an upper bound on the renewal duration ρ_{\max} . Thereby, the PCA sweeps values between $0 \leq \chi \leq \rho_{\max}$, and returns the best solution (i.e., the one that maximizes U). We denote the discovery rate that the PCA obtains by U_A and the configuration parameters it returns by (λ_A, l_A) .

3.5.3 An Upper Bound

To compute an *upper bound* on Panda's optimal discovery rate, we first derive a *lower bound* on the discovery power in an optimal solution, denoted by $\underline{\Phi}^*(1)$.

Theorem 1. *The discovery power in an optimal solution, $\Phi^*(1)$, satisfies,*

$$\Phi^*(1) \geq \frac{U_A}{N} (P_r M + C_{rs} + C_{sr}) := \underline{\Phi}^*(1), \quad (3.13)$$

where U_A is the discovery rate returned by the PCA.

Proof. The discovery energy for the optimal solution can be rewritten as

$$\begin{aligned} \Phi^*(1) = & \underbrace{\frac{(N-1)(1-e^{-\lambda^* l^*}) P_r \chi}{N \rho}}_{\geq 0} + \\ & \underbrace{\frac{(N-1)(1-e^{-\lambda^* l^*})}{\rho}}_{=U^*} \frac{P_r M + C_{rs} + C_{sr}}{N}, \end{aligned}$$

where l^* and λ^* are an optimal configuration. By definition of the optimal solution, $U^* \geq U_A$ for any feasible configuration parameters leading to a discovery rate of U_A . Since, $\chi \geq 0$, the proof follows. \square

Using (3.10) and (3.13), an upper bound optimization problem is formulated as,

$$\max_{\lambda, l} \quad \overline{U}^* = (N-1)\lambda l / \rho \quad (3.14)$$

$$\text{s.t.} \quad \Phi(0) + \underline{\Phi}^*(1) \leq P_b. \quad (3.15)$$

Note that (3.15) effectively considers only a portion of the discovery power. This implies that the upper bound solution may be infeasible as it will incur an average power spending value higher than P_b . However, a solution maximizing (3.14) is in fact an upper bound on the optimal solution U^* , and is denoted by \overline{U}^* .

To obtain an optimal solution for the upper bound \overline{U}^* , we first solve for λ with respect to the power constraint in (3.15):

$$\lambda = P_b / (C_{rs} + C_{st} + P_r l + P_t M - N P_b (M + l)) \quad (3.16)$$

Solving $d\overline{U}^*/dl = 0$ obtains the listen time to maximize \overline{U}^* ,

$$l = \frac{\sqrt{(P_r (P_r - N P_b) (C_{rs} + C_{st} + P_t M) (C_{rs} + C_{st} + P_t M - M N P_b))}}{P_r^2 - N P_r P_b}$$

Going forward, the corresponding value of the upper bound will be referred to as \overline{U}^* .

3.5.4 Performance of the PCA

We now compare the discovery rate from the PCA (U_A) to the upper bound discovery rate computed in Section 3.5.3 ($\overline{U^*}$). In this section, we will refer to the ratio $U_A/\overline{U^*} \leq 1$ as the *approximation ratio*. Values close to 1 imply that U_A is close to $\overline{U^*}$, and therefore, also close to the true optimal U^* .

Recall that the upper bound is computed by ignoring part of the discovery power, and therefore, violating the power constraint. Hence, when the discovery power $\Phi(1)$ in the optimal solution is indeed negligible (≈ 0), the upper bound $\overline{U^*}$ is close to the true optimal, U^* . Therefore, as the discovery power decreases, the approximation ratio approaches 1.

In Fig. 3.4 we show both the discovery power and the approximation ratio resulting from the configuration parameters returned by the PCA for varying $N \in \{2, 5, 10, 25\}$. In Fig. 3.4(a), we consider the discovery power as a proportion of the total power budget P_b . As shown, smaller values of N or P_b result in a smaller proportion of discovery power.

Fig. 3.4(b) shows the approximation ratio $U_A/\overline{U^*}$ as a function of the power budget P_b . First, note that the approximation ratio is always greater than 94% for all parameters considered. Therefore, because $U_A \leq U^* \leq \overline{U^*}$, the discovery rate provided by the PCA is within 6% of the optimal. Additionally, for larger values of N or P_b , the approximation ratio decreases. In this domain, the discovery power is larger (see Fig. 3.4(a)), and therefore, the discovery rate returned by the PCA (U_A) is actually closer to the optimal (U^*).

3.6 Panda-Dynamic (Panda-D)

Panda is analyzed assuming that nodes are homogenous (A1), are arranged in a clique (A2), and the number of nodes N is known a priori (A3). However, when these assumptions do not hold, the expected power consumption of a node operating with Panda (see Section 3.4.4) will vary and the power budget is no longer satisfied. Therefore, in this section, we present Panda-Dynamic (Panda-D).

Panda-D operates with the same behavior as Panda, transitioning between the sleep, receive, and transmit states. However, to handle the varying power consumption with the relaxed

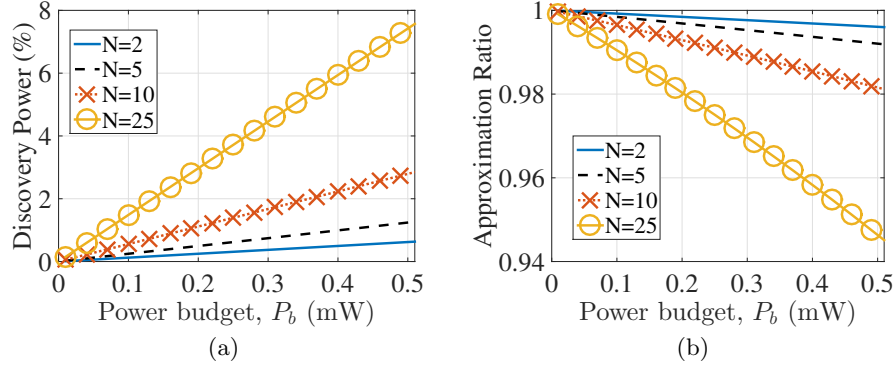


Figure 3.4: The performance of the PCA for varying power budgets, P_b , and number of nodes, N : (a) discovery power as the percentage of the power budget ($\Phi(1)/P_b$), and (b) the approximation ratio (U_A/\overline{U}^*).

assumptions, the rate of the exponential sleep duration is dynamic, and is adapted based on the voltage of the capacitor,⁷ V_{cap} . Thereby, if a node consumes too much power, its voltage will decrease and it will adapt by staying in the sleep state for longer durations.

Formally, the configuration parameters for Panda-D are computed as follows. In this case, P_b represents an *estimated* power budget for each node, yet we allow for each node to harvest power at varying rates around P_b . The sleep duration is scaled such that the nodes' anticipated power consumption is 0.01mW when $V_{\text{cap}} = 3.6\text{V}$, and is P_b when $V_{\text{cap}} = 3.8\text{V}$. From the two points, the desired power consumption of the node, P_{des} , is computed as a linear function of the capacitor voltage (V_{cap}),

$$P_{\text{des}}(V_{\text{cap}}) = \frac{P_b - 0.01}{3.8 - 3.6} (V_{\text{cap}} - 3.6) + 0.01, \quad 3.6 \leq V_{\text{cap}} \leq 4.$$

Based on the desired power consumption P_{des} , the node adjusts its sleep duration. As mentioned above, we cannot explicitly relate the sleep duration to the power consumption for a node. Instead, we will estimate the power consumption by ignoring the *discovery power*. That is, we assume that a node always follows the sleep, receive, transmit cycle and is spending on

⁷A similar adaptation mechanism was also proposed in [159].

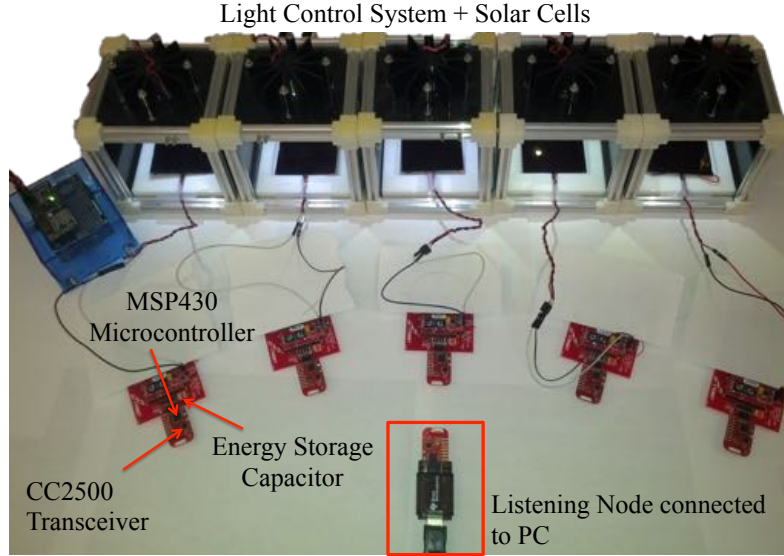


Figure 3.5: Panda experimental setup: 5 EH nodes harvest energy through the solar cells with neighbor discovery rates monitored by a listening node.

average at rate,

$$P_{\text{est}} = \frac{\eta(0)}{1/\lambda + l + M} = \frac{P_r l + P_t M + C_{sr} + C_{ts}}{1/\lambda + l + M}.$$

The average sleep duration, $1/\lambda$, is computed as a function of the capacitor voltage (V_{cap}) by solving $P_{\text{est}} = P_{\text{des}}$,

$$\frac{1}{\lambda} = \frac{P_r l + P_t M + C_{sr} + C_{ts}}{P_{\text{des}}(V_{\text{cap}})} - l - M. \quad (3.17)$$

We remark that the listen time l is obtained using the PCA with $N = 2$ (i.e., we try to maximize the discovery rate for each *directional link*).

We claim that the robustness of Panda-D is two-fold. First, it is power aware and thus nodes can operate under different and varying power harvesting rates, relaxing (A1). Additionally, it does not require any a priori knowledge of the size or topology of the network, relaxing (A2) and (A3).

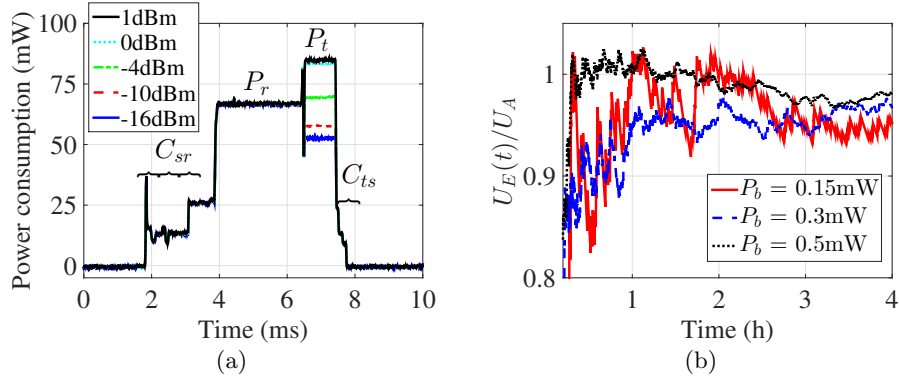


Figure 3.6: (a) Power consumption and transition costs for different transmission power levels for a node transitioning between the sleep, receive, and transmit states, and back to sleep. (b) Panda experimental evaluation with varying power budgets, P_b : Convergence of the experimental discovery rate (U_E) to the analytical discovery rate (U_A) for $N = 5$.

3.7 Experimental Performance Evaluation

We now evaluate Panda using a testbed, pictured in Fig. 3.5, composed of TI eZ430-RF2500-SEH [5] prototypes (described in Section 3.3.1). First, we evaluate Panda in the context of the model presented in Section 3.3.2. We compare Panda’s experimental discovery rate, denoted by U_E , to related work. Additionally, we present Panda’s performance with varying parameters (e.g., transmission power, message length). Then, we evaluate Panda-D in scenarios with non-homogeneous power harvesting and multihop topologies.

3.7.1 Protocol Implementation

In accordance with Panda, the microcontroller builds the discovery message and sends it to the low-power transceiver. Table 3.2 illustrates the structure of the discovery message. The message contains debugging information, the source ID of the transmitting node, and the node’s capacitor voltage (which is sampled from the ADC). Additionally, the message includes the number of discoveries from each neighbor since the initialization of the experiment, referred to as the node’s *neighbor table*. The total length of a discovery message is 18 bytes and the resulting transmission duration of the discovery message is 0.92ms.

In order to characterize the energy costs, we measure the power consumption of the microcontroller and transceiver using an oscilloscope. Fig. 3.6(a) shows the power levels for a node transitioning between the sleep, receive, and transmit states. We compute the average power consumption and transition energy for each state, with values summarized in Table 3.3.

We note that the transition times to and from the sleep state are non-negligible (in some cases a few ms). To account for this, these transition times are considered as part of the sleep state and, are therefore, subtracted from the actual sleep duration. We elaborate further on the importance of incorporating these switching costs in Appendix 3.9.4.

The parameters in Table 3.3 compose the inputs to the PCA, which computes the rate of the exponential sleep λ_A and the duration of the listen state l_A as well as an expected discovery rate U_A . These *configuration parameters* are loaded into the nodes for experimental evaluation in which we observe the discovery rate as well as the power consumption.

3.7.2 Testbed and Experimental Setup

We consider networks of 3, 5, and 10 nodes ($N = 3, 5, 10$). We consider power budgets of $P_b = 0.15, 0.3, 0.5\text{mW}$; these are aligned with other solar harvesting budgets [69]. Initially, to confirm the practicality of Panda when assumptions (A1), (A2), and (A3) hold, we place the nodes in close proximity with a homogenous power budget. In Section 3.7.7, we will evaluate Panda-Dynamic (Panda-D) and relax these assumptions by considering a multihop topology and non-homogenous power harvesting.

To facilitate experimental evaluation with up to $N = 10$ nodes, in addition to an EH node shown in Fig. 1.6(b), we also incorporate nodes powered by AAA batteries into the experiments. Both the EH node and the node powered by AAA batteries operate using the same configuration parameters and hence have identical behaviors. However, we carefully logged the power consumption of the EH node by including control information in the discovery message (see Table 3.2).

We utilize a listening node consisting of a microcontroller and transceiver set to a promiscuous sniffing mode to log experimental results. Powered by a USB port on a monitoring PC, the listening node reports all received messages to the PC for storage and post processing. The

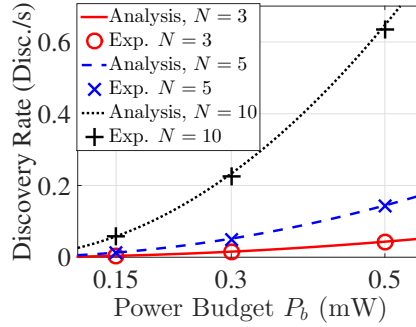
Byte	Data
0	Packet length (18)
1	Type
2-11	Neighbor table
12-13	Capacitor voltage
14-15	Debugging information
16	Transmissions counter
17	Originating node ID

Table 3.2: Discovery message structure.

experimental discovery rate, U_E , is computed by dividing the total number of discoveries since the initialization of the experiment by the experiment duration. Clearly, the time until which the experimental discovery rate converges depends on the rate of discovery. In Fig. 3.6(b), we observe the experimental discovery rate, U_E over time for $N = 5$ and $P_b = 0.15, 0.3, 0.5\text{mW}$. Based on the results shown in Fig. 3.6(b), all experiments were conducted for up to 96 hours.

The light levels are set to correspond to each of the power budgets, P_b . However, the performance of the solar cells vary significantly due to external effects such as aging, orientation, and temperature [61]. To mitigate these affects and facilitate repeatable and controllable experiments, we utilize a software controlled light system from [112] which we describe in Appendix 3.9.5.

Additionally, as mentioned in Section 3.3, the prototype is not power aware. That is, although we can accurately measure the power harvested by the solar cell, it is difficult to control the energy actually stored in the capacitor, due to numerous inefficiencies of the harvesting circuitry, which are further described in Appendix 3.9.5. As such, we empirically estimated the harvesting inefficiency to be 50% and adjust the light levels to provide each node energy according to the value of P_b chosen.

Figure 3.7: Panda's discovery rate with varying power budgets, P_b , and number of nodes, N .

Param.	P_t	P_r	M	C_{sr}	C_{rs}	C_{tx}
Value	59.23mW	64.85mW	0.92ms	74.36μJ	13.48μJ	4.83μJ

Table 3.3: Measured prototype parameters.

3.7.3 Discovery Rate

For each (N, P_b) pair, we evaluate Panda, with the experimental parameters summarized in Table 3.4. First, we note that Panda's duty cycle is typically between 0.1–0.6%, which is significantly lower than the duty cycles considered in related protocols [148]. Additionally, note the accuracy of the analytical discovery rate, U_A , computed from (3.2), compared to the experimental discovery rate, U_E . On average, the error between them is $\approx 1\%$. This confirms the practicality of Panda and the model described in Section 3.3.

In Fig. 3.7, we plot the experimental and analytical discovery rate for each value of (N, P_b) shown in Table 3.4 and observe the effect of varying N and P_b . As expected, the discovery rate increases as P_b increases. The number of nodes N is directly correlated with the discovery rate, as indicated in (3.2) and (3.3). As such, the discovery rate increases as N increases.

To further ensure that the nodes behave as expected, we tracked the neighbor table in every node using the transmitted discovery messages. Table 3.5 shows the nodes neighbor tables for $P_b = 0.3\text{mW}$. The table shows that all nodes are able to discover one another and exhibit similar discovery rates.

N	P_b (mW)	λ_A^{-1} (ms)	l_A (ms)	Duty Cycle (%)	U_A (Disc./s)	U_E (Disc./s)	Error (%)	Run Time (h)
3	0.15	1778.68	2.066	0.168	.0039	.0038	-1.35	36
	0.3	887.39	2.070	0.336	.0156	.0154	-1.23	36
	0.5	530.88	2.075	0.561	.0434	.0438	1.07	48
5	0.15	1777.18	2.068	0.168	.0130	.0132	1.43	96
	0.3	885.91	2.075	0.337	.0519	.0518	-0.33	60
	0.5	529.43	2.084	0.564	.1443	.1427	-1.15	18
10	0.15	1773.49	2.075	0.169	.0584	.0589	0.89	18
	0.3	882.32	2.089	0.340	.2332	.2341	0.38	18
	0.5	525.97	2.107	0.572	.6470	.6510	0.62	18

Table 3.4: Panda experimental parameters: (λ_A, l_A) generated using the PCA for every input (N, P_b) pair and the resulting analytical (U_A) and experimental (U_E) discovery rate.

3.7.4 Discovery Latency and Comparison to Related Work

The discovery latency is the time between consecutive discoveries for a *directional link*. It can be an important parameter for numerous applications where nodes are only within communication range for short periods of time. Although the objective of Panda is to maximize the discovery rate, in Fig. 3.8(a), we show the CDF of the discovery latency for each *directional link* in an experiment with $N = 5$ and varying power budgets. Clearly, the average discovery latency decreases as the average discovery rate increases. Thus, for a higher power budget, the discovery latency decreases.

Previous work [24, 53, 149] focused on minimizing the worst case discovery latency for a link. We compare the discovery latency of Panda, shown in Fig. 3.8(a), to previous work. However, as mentioned in Section 3.2, previous work considers a duty cycle constraint instead of a power budget (P_b) . To provide a means of comparison, we use the following equation to relate the power constraint to a duty cycle.

$$P_b = \text{Duty Cycle}(\%) \cdot \text{Average Active Power (mW)} \quad (3.18)$$

ND Table	1	2	3	4	5	Total RX
1	0	35	42	32	42	152
2	24	0	23	45	38	130
3	39	36	0	21	33	129
4	36	35	46	0	32	149
5	38	42	42	42	0	164

Table 3.5: Neighbor table for $N = 5$, $P_b = 0.3\text{mW}$ after 4 hours. Entry (i,j) shows the number of discoveries of node j by node i .

We compare to the deterministic Searchlight protocol [24], which minimizes the worst case discovery latency [149]. We also compare to the well-known probabilistic Birthday (BD) protocol [115]. To account for the power budget, we modify these protocols based on (3.18) (with details explained in Appendix 3.9.6) and denote them as Searchlight-E and BD-E. Based on previous work [149], we set the slot size for Searchlight-E and BD-E to 50ms and add an overflow guard time of 1ms.

In Fig. 3.8(b), we compare the average discovery rate for Panda vs. simulations of the Searchlight-E and BD-E protocols. We found that Panda typically outperforms the Searchlight-E and BD-E protocols by over 3x in terms of the average discovery rate.⁸

Furthermore, in Fig. 3.8(c), we consider the *worst case* discovery latency and show that although Panda has a non-zero probability of having any discovery latency, for the experiments we considered, the 99th percentile of discovery latency outperformed the Searchlight-E protocol worst case bound by up to 40%.

Note that the Searchlight protocol was proven to minimize the worst case discovery latency. However, as shown through our evaluation, Panda outperforms Searchlight-E by a factor of 3x in terms of average discovery rate. Moreover, in most cases (over 99%), the discovery latency is below the worst case bound from Searchlight-E. This emphasizes the importance of incorporating

⁸As described in Appendix 3.9.6, the simulations of Searchlight-E and BD-E do not account for packet errors or collisions. As such, the discovery rates for these protocols is likely to be lower in practice.

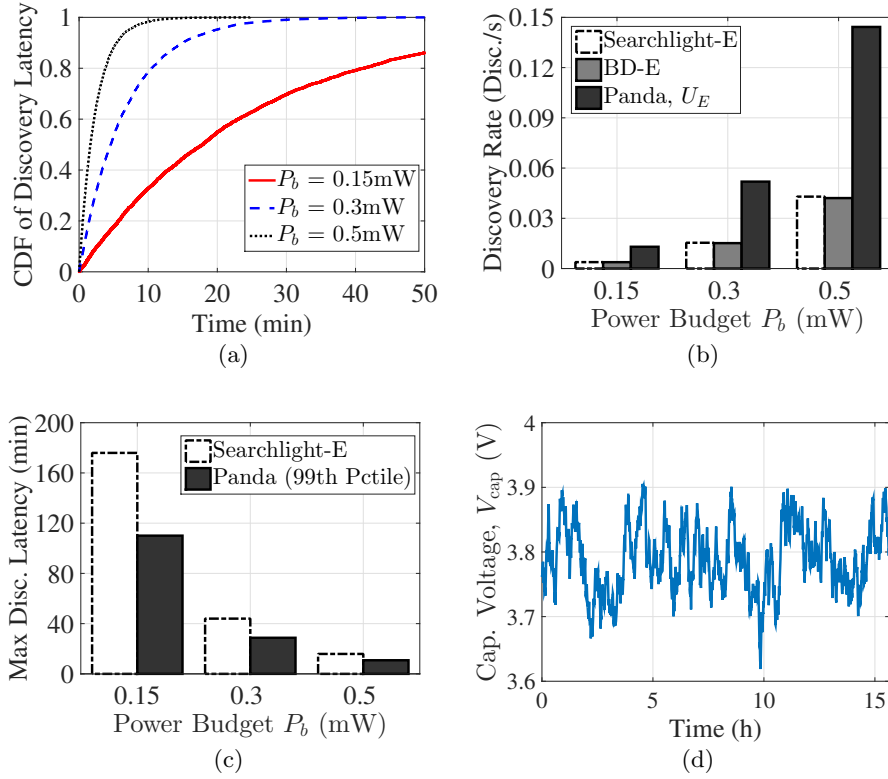


Figure 3.8: Panda experimental evaluation for $N = 5$: (a) CDF of per link discovery latency; comparison to SearchLight-E [24] and BD-E [115] of (b) the discovery rate and (c) the worst case latency; (d) capacitor voltage for a node with a 30mF capacitor and $P_b = 0.5\text{mW}$.

a detailed power budget, as is done in Panda, as opposed to a duty cycle constraint.

3.7.5 Power Consumption

Using Panda, a node consumes power at a rate of up to P_b (mW), *on average*. However, the power consumption is stochastic, and therefore, it is expected that the energy stored will vary over time. In Fig. 3.8(d), we show the capacitor voltage over time for a node with $N = 5$ and $P_b = 0.5\text{mW}$. Energy neutrality is demonstrated by the oscillation in the energy level within the limits of the capacitor storage. Recall from Section 3.3 that if the energy drains below a software induced threshold of 3.6V, the node temporarily sleeps for 10s to regain energy. These periods of additional sleep affect the discovery rate and, as indicated by the accuracy of the

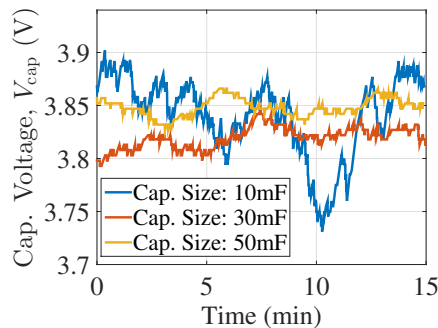


Figure 3.9: Panda performance evaluation for $N = 5$, $P_b = 0.15\text{mW}$: Voltage level of nodes with varying capacitor sizes over 15 minutes.

experiments, these occurrences are rare.

Furthermore, in Fig. 3.9, we experiment with varying capacitor sizes ranging from 10-50mF. As expected, smaller capacitors have added variation in the voltage level. Therefore, smaller capacitors can reach the upper (fully charged) or lower (empty) voltage limits more frequently than larger capacitors. In practice, the capacitor should be sized with respect to the variation in the power consumption and power harvested.

3.7.6 Panda Design Considerations

We now consider Panda's performance for varying transmit power and discovery message durations.

3.7.6.1 Transmit Power, P_t

The transmission power can be set in software. A larger transmission power can result in more geographical coverage, but also consumes more energy. In Fig. 3.10(a), we consider $N = 5$ and $P_b = 0.5\text{mW}$ and observe how the discovery rate changes with varying transmission powers. A larger transmission power requires nodes to sleep longer before transmitting, resulting in less discoveries. Note that for this experiment the energy costs from Table 3.3 no longer hold and we remeasured them to compute the configuration parameters.

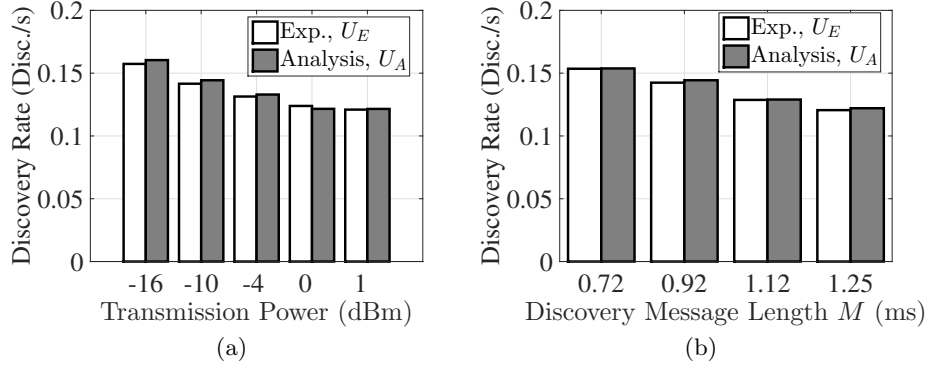


Figure 3.10: Panda performance evaluation for $N = 5$ and $P_b = 0.5\text{mW}$: Experimental and analytical discovery rates under varying (a) transmission power (P_t) and b) discovery message length (M).

3.7.6.2 Discovery Message Duration, M

The discovery message requires M ms to be transmitted and contains the node ID and neighbor table information. By adjusting the modulation/coding of the radio or the data content, the packet length can be shortened. A shorter packet length results in less time transmitting as well as less time listening for messages. As shown in Fig. 3.10(b), smaller packet sizes result in an increase in the discovery rate. This presents an application design decision if the contents of the packet can be adjusted to obtain a desired discovery rate.

3.7.7 Panda-Dynamic

We now evaluate Panda-D (described in Section 3.6). The only input to Panda-D is the estimated power harvesting rate, $P_b = 0.15\text{mW}$, and the capacitor voltage V_{cap} . From (3.17), the average duration of the exponential sleep is then computed as,

$$\frac{1}{\lambda} = \frac{382.2238}{V_{\text{cap}} - 3.5857} - 2.9843 \text{ (ms)}. \quad (3.19)$$

Thus, the node scales its power consumption based on V_{cap} . For example, at $V_{\text{cap}} = 3.6\text{V}$ and 4V , the node will sleep on average for 26.75 and 0.92 seconds, respectively.

To estimate the average sleep duration for a given node in Panda-D, we compute the average value of V_{cap} over the course of an experiment. Based on this value, the average sleep

duration is estimated from (3.19).

Panda-D does not require a priori information of the number of neighbors, N . Therefore, throughout this section, (A3) is relaxed. Below, we observe the performance of Panda-D first when (i) nodes remain in a clique topology with homogenous power budgets. Then we consider Panda-D (ii) in a multihop topology (relaxing (A2)), and finally (iii) in non-homogenous power harvesting scenario (relaxing (A1)). Relaxing all assumptions together requires running a live real-world experiment and is a subject of future work.

(i) Comparison to Panda: We first evaluate Panda-D with an experimental setup similar to the one shown in Fig. 3.5. Specifically, we consider a network of $N = 3$ nodes in close proximity with a power harvesting rate of $P_b = 0.15\text{mW}$.

As shown in Fig. 3.11(a), the capacitor voltage for all 3 nodes stays approximately near 3.8V. As described in Section 3.6, the average power consumption at 3.8V is approximately P_b . Therefore, in this scenario, Panda-D and Panda have similar power consumption and discovery rates. As such, the experimental discovery rate of Panda-D is within 1% of the analytical estimate of Panda.

(ii) Multihop Topologies: Previously, we assumed that all nodes form a clique topology with no packet losses (A2) and the number of nodes N known (A3). Indeed, for the experiments conducted above with a transmission power of -10dBm , we found that nodes within $\approx 20\text{m}$ could be treated as a clique topology with over 99% packet success rates.

However, to evaluate a non-clique topology and relax (A2) and (A3), we manually reconfigured the transmission power to -26dBm and set 3 nodes in a line topology with distance between nodes 1-2 and 2-3 of 1.5m, as shown in Fig. 3.11(b). In this configuration, nodes rarely receive messages from their two-hop neighbors. Nodes run Panda-D and are given light levels corresponding to the power harvesting rate of $P_b = 0.15\text{mW}$ (as described in Section 3.7.2). After 50 hours, the resulting discovery rate is shown on each link in Fig. 3.11(b).

The two extreme nodes (nodes 1 and 3) have very few discoveries from one other, due to the distance between them. However, the node in the middle (node 2) forms an effective clique of size 2 with each of its neighbors. We therefore can analyze the discovery rate per link. For example, the discovery rate of the link between nodes 1 and 2 is 0.0051 disc./s, which is within

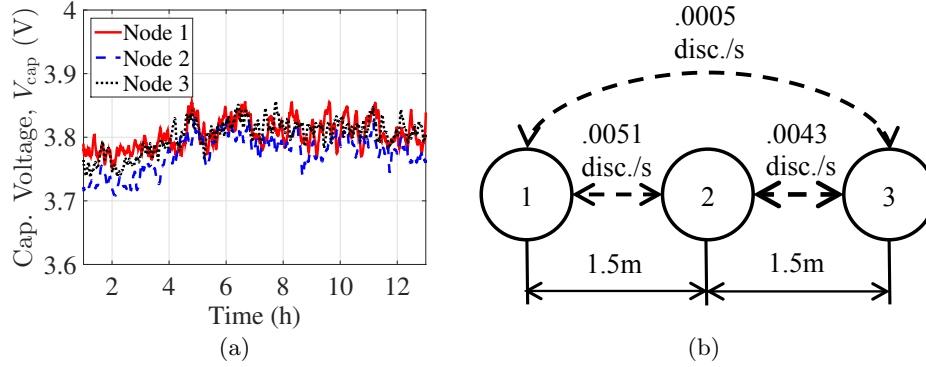


Figure 3.11: Panda-D experimental evaluation for $N = 3$ and homogeneous power harvesting $P_b = 0.15\text{mW}$: (a) Capacitor voltage, V_{cap} , in a clique topology. (b) Per link experimental discovery rates for a line topology after 50 hours.

1% of the analytical discovery rate for a clique with $N = 2$ and $P_b = 0.15\text{mW}$. Therefore, even with non-clique topologies, each link that is within communication range can be analyzed as a network with $N = 2$. This implies that issues such as the hidden-node problem do not significantly affect the performance of Panda.

(iii) *Non-Homogeneous Power Harvesting:* We now consider nodes 2–5 using Panda-D with light levels corresponding to power harvesting of 0.075, 0.15, 0.225, 0.3mW, respectively. Node 1 is a *control* node running Panda with $P_b = 0.15\text{mW}$ and $N = 5$.

For each of the 4 Panda-D nodes, the capacitor voltage, V_{cap} , is shown in Fig. 3.12(a) and settles based on the power harvesting. Variations in the settling voltage stem from the dynamic average sleep duration at different power harvesting levels. For example, node 5 is given a light level of 0.3mW, and therefore, has a shorter sleep duration than node 2 (light level of 0.075mW). Correspondingly, Fig. 3.12(b) shows the neighbor table: entry (i,j) represents the number of discoveries of node j by node i over the experiment duration. Due to non-homogeneity, the discovery rate for each link depends on the power harvested; nodes with larger power budgets discover their neighbors, and are discovered, more frequently.

In Appendix 3.9.7, we treat each link with non-homogenous power harvesting as a clique ($N = 2$), and estimate its discovery rate; the approximation is within 20% of the experimental value.

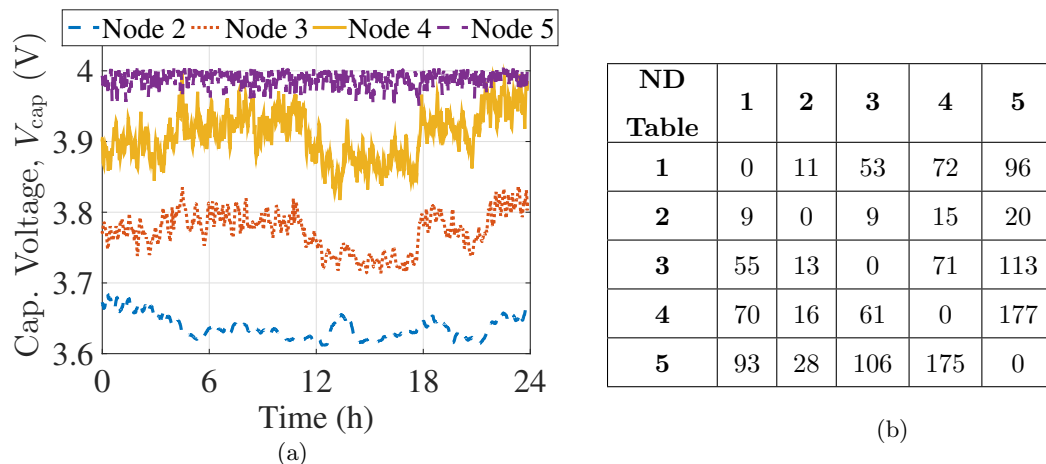


Figure 3.12: Panda-D experimental evaluation for non-homogeneous power harvesting experiment with $N = 5$ over 24 hours: (a) Capacitor voltage, V_{cap} , and (b) resulting neighbor table.

3.8 Conclusions and Future Work

We designed, analyzed, and evaluated Panda, an ND protocol for EH nodes. By accounting for specific hardware constraints (e.g., transceiver power consumption for transmission, reception, and state switching), Panda obeys a *power budget*. Using renewal theory, we developed the Panda Configuration Algorithm (PCA) to determine the node's sleep and listen durations which maximize the discovery rate; the PCA achieves a nearly-optimal discovery rate (over 94%).

We evaluated Panda using TI eZ430-RF2500-SEH EH nodes. The real-life accuracy was consistently within 2%, demonstrating the practicality of our model. Furthermore, Panda outperformed the closest related protocols Searchlight-E [24] and BD-E [115] by achieving a discovery rate that was up to 3x higher. Finally, we showed that a version of the protocol, Panda-Dynamic, was able to adapt to scenarios with non-homogeneous power harvesting and multihop topologies.

Panda can be readily applied for nodes with a non-rechargeable battery, where the power budget is set based on the desired lifetime. Future work will consider relaxing additional assumptions of our model. Primarily, we will attempt to *optimize* Panda-D in the presence of nodes with heterogenous power budgets in non-clique topologies. Additionally, we will consider alternate formulations to achieve closed form optimal configuration parameters. Finally, we will

consider a protocol with sleep duration distributions other than exponential.

3.9 Appendices

3.9.1 Modifications to the TI eZ430-RF2500-SEH [5]

The modifications can be found in Table 3.6.

Table 3.6: Modifications to the TI eZ430-RF2500

Component	Problem/Modification
Energy Storage	On-board battery cannot be monitored; disable on-board battery and replace with an external capacitor.
Solar Panel	On-board solar cell cannot be monitored; disable on-board cell, measure power harvested by connecting a multimeter in series with solar cell from [137].
Power Consumption	Unable to track power consumed; measure consumed power with an oscilloscope across a 10Ω sense resistor, placed in series with the transceiver and the microcontroller.
12kHz Clock Source	Clock frequency varies by up to 20% for each node; rectified by manually measuring/calibrating the number of clock ticks in one second for each device.

3.9.2 Incorporating Idle Power Consumption

In this work, we disregard the power cost of nodes in the sleep state. In this section, we explain how these costs can be incorporated. As described in Section 3.7, the idle cost of the microcontroller consumes $1.6\mu\text{W}$. This draw is constant for all states (sleep, listen, and idle). As such, to incorporate it into our model, it is simply subtracted from the power budget P_b .

In Section 3.4.4, we ignore the expected amount of energy (μJ) consumed by a node in the sleep state ($y = 2$). The only energy consumed by such a node occurs when it begins to listen while a packet is currently being transmitted (exemplified by Node 5 in Fig. 3.3). In this case, the node spends energy to transition to and from the sleep state, as well as listen for a short fixed Clear Channel Assessment (CCA) period, denoted as t_{CCA} . The energy consumption is

then $\eta(2) = C_{sl} + P_r t_{CCA} + C_{ls}$.

This event occurs in a renewal with probability given by

$$\Pr(Y = 2) = \frac{N-1}{N} (e^{-\lambda l})(1 - e^{-\lambda M}).$$

In words, first the node must not be the transmitter w.p. $(N-1)/N$. As the node is asleep when the transmitter begins to listen, it must then sleep for at least l duration. Finally, given that it is in the sleep state when the transmitter begins to transmit, it must then wakeup before the message is transmitted (duration M).

Thus for our experimental evaluation in Section 3.7, the idle power costs are summarized in Table 3.7 for the experimental parameters originally presented in Table 3.4. As can be seen, $\Pr(Y = 2)$ is very small implying that it is quite rare that a node wakes up in the middle of a transmitted packet. As such, the percentage of the power budget consumed, on average, is always less than 0.5%, and therefore can be ignored. We note, however, that the PCA can easily be modified to incorporate this added power consumption.

N	P_b (mW)	$\Pr(Y = 2)$	Expected Energy per Renewal (μ J) ($\eta(2) \cdot \Pr(Y = 2)$)	Pct of P_b (%)
3	0.15	0.34e-3	0.0302	0.034
	0.3	0.69e-3	0.0605	0.068
	0.5	1.15e-3	0.1010	0.112
5	0.15	0.41e-3	0.0363	0.068
	0.3	0.83e-3	0.0728	0.135
	0.5	1.38e-3	0.1215	0.223
10	0.15	0.47e-3	0.0410	0.151
	0.3	0.94e-3	0.0822	0.300
	0.5	1.57e-3	0.1376	0.495

Table 3.7: Panda idle power consumption for every input (N, P_b) pair from Table 3.4: the probability of a node waking in the middle of a packet ($\Pr(Y = 2)$) and the expected portion of the power budget (P_b) consumed.

3.9.3 Derivation of χ

The derivation of χ is as follows. Without loss of generality, let us assume that the transmitter in a renewal (\mathcal{N}_t , e.g., node 6 from Fig. 3.3) enters the listen state at $t = 0$, and at $t = l$, it transmits the discovery message. Let x denote the idle listening time for a given node where x is exponentially distributed with $0 < x < l$. We look to find $\chi = \mathbb{E}[x|x < l]$, i.e.,

$$\begin{aligned}\chi &= \int_{t=0}^{+\infty} \Pr(x > t|x < l) dt \\ &= \int_{t=0}^l \frac{(1 - e^{-\lambda t}) - (1 - e^{-\lambda l})}{1 - e^{-\lambda l}} dt = \frac{1}{\lambda} - \frac{le^{-\lambda l}}{1 - e^{-\lambda l}}\end{aligned}$$

3.9.4 Importance of Switching Costs

In this work, we incorporate the costs to switch to and from different radio states (sleep, receive, transmit). In this section, we demonstrate the importance of accounting for these costs (which are commonly overlooked in related work).

As the PCA allows for arbitrary switching costs, in Table 3.8, we compute the parameters assuming that $C_{ij} = 0$ ($\forall i, j \in \{s, r, t\}$). As indicated in the table, the discovery rate improves by 2-3x compared to the discovery rate when including the switching costs from Table 3.3. However, the power consumed by transition causes the power budget to be exceeded by up to 80%. Therefore, ignoring the switching costs may improve the discovery rate, but also results in significantly more power consumed.

3.9.5 Software Controlled Light System and Harvesting Inefficiencies

We utilize an advanced software controlled light system from [112] (shown in Fig. 3.13) that uses a Java-based script and Arduino-based light control modules to precisely control the irradiance (light energy intensity) generated by LEDs. The system can produce 1024 distinct irradiance levels between 0 and 14mW/cm² and the irradiance levels can be changed with time steps of under 100ms. Dark box enclosures and 3D printed mounting fixtures ensure full control over the light conditions at the solar cells. This guarantees that our experimental evaluations *are based on the same energy inputs*.

N	P_b (mW)	U_A , w/ Switching Costs (disc./s)	U_A , w/o Switching Costs (disc./s)	Power Consumed (mW)
3	0.15	0.0039	0.010	0.26
	0.3	0.0156	0.038	0.52
	0.5	0.0434	0.107	0.86
5	0.15	0.0130	0.032	0.26
	0.3	0.0519	0.128	0.52
	0.5	0.1440	0.359	0.87
10	0.15	0.0584	0.144	0.26
	0.3	0.2330	0.581	0.52
	0.5	0.6470	1.630	0.87

Table 3.8: Panda performance evaluation: the discovery rate U_A resulting from the PCA, and the actual power consumed when ignoring the switching costs.

Furthermore, we conduct extensive experiments utilizing a UV818 photodetector to carefully calibrate the irradiance of the light control system. We characterize the power harvested under both the ambient light and the software controlled light setup. The measurement setup is shown in Fig. 3.14(a). We connect the solar cell in series with a multimeter to measure current. The voltage of the solar cell is at 1.02V and hence the harvested power can be easily calculated. Then, by sweeping the irradiance using the software controlled light system, Fig. 3.14(b) shows the power harvested for a range of up to 1mW. Using the mapping found in this characterization, we can control the power harvested by the solar cell.

However, the actual power that is stored depends on numerous inefficiencies in the power harvesting circuitry. Specifically, the node contains a Cymbet CBC5300 to up-convert the power harvested from the solar cell at 1.02V to the capacitor charging voltage of 4V, which consumes some overhead energy. In addition, there are inefficiencies in the regulation circuit which regulates an output voltage of 3.5V to power the load. These inefficiencies are difficult to characterize as they vary based on uncontrollable external factors such as the temperature and component variations.

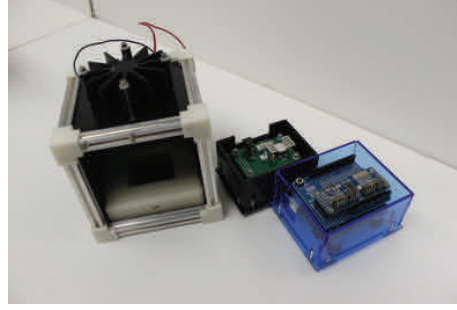


Figure 3.13: Software controlled light system (originally designed in [112]) including a dark box enclosure, high-power LED driver, and an Arduino-based light controller.

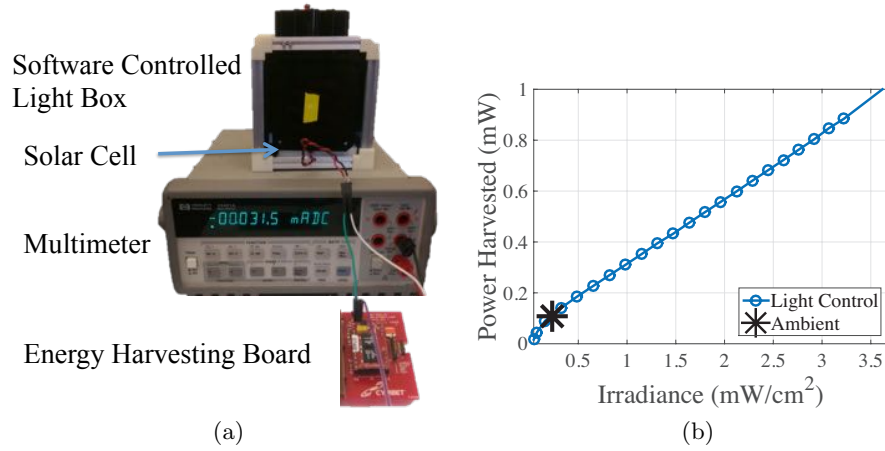


Figure 3.14: Energy harvesting characterization: (a) measurement setup and (b) resulting power harvested as a function of the irradiance produced by the software controlled light system.

In our evaluations (see Section 3.7), nodes were given light levels which corresponded to their power budget P_b . Due to the inefficiencies described above, setting the light levels to correspond to each power budget for each node was difficult.

To accomplish this, we conduct a 4-day experiment in which nodes operated using Panda, yet we varied the light levels every 6 hours. An example of the capacitor voltage for one node in this experiment is shown in Fig. 3.15. Each valley represents a 10-minute “dark” period where the light is completely off before changing to the next light levels.

With limited light levels (i.e., hours 0-20 in Fig. 3.15), the capacitor voltage operates near the minimum implying that the node is consuming more energy than it harvests. With larger

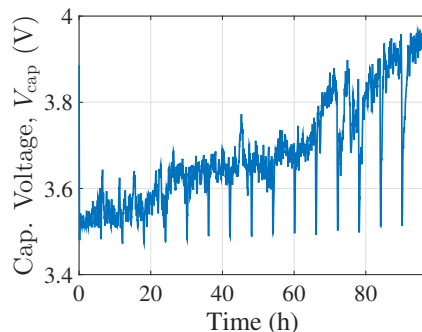


Figure 3.15: A four day experiment to find the light level where the node stays around its energy neutral point.

light levels (i.e., hours 80-100), the node is harvesting more energy than it consumes and thus the capacitor voltage reaches its upper limit. However for the range of lights corresponding to 20-80 hours in Fig. 3.15, the node has a relatively stable voltage, implying that it is consuming power (on average) as the same rate it harvests; *energy neutrality* is obtained.

By performing this experiment for all nodes, we found the light levels at which each node is energy neutral. The neutral light levels varied significantly. Furthermore, by comparing the power harvested by the solar cell to the power budget (P_b), we found that the efficiency of the storage process to be between 40% and 60%. This emphasizes the need to incorporate energy storage feedback into the ND protocol, as is done by Panda-D.

3.9.6 Description of Searchlight-E and BD-E

In this appendix, we describe our adaptations to two related protocols, termed Searchlight-E [24] and Birthday-E (BD-E) [115]. Both of these protocols are based on nodes maintaining time slots. We will denote the time slot duration as d_s . In each *active* time slot, a node sends a beacon message (originally proposed by [53]). The beacon begins a slot with a packet transmission, then listens to the channel, and ends the slot with a packet transmission.

In the Searchlight protocol, two slots are active per cycle of length t slots. Therefore, the power budget can be written as

$$\frac{2(2P_t M + P_r(d_s - 2M) + C_{st} + C_{ts})}{td_s} \leq P_b.$$

Table 3.9: Panda-D Experimental and Predicted Parameters for the $N = 5$ experiment from Fig. 3.12.

	1	2	3	4	5
Exp. Avg. Voltage (V)	3.5142	3.6432	3.7772	3.9122	3.9880
Exp. Avg. Sleep Duration, $1/\lambda_i$ (ms)	1777.2	7542.6	2041.2	1181.5	947.9
Est. Avg. Sleep Duration, $1/\lambda_i$ (ms)	1777.2	6644.4	1993.0	1167.7	947.1

Table 3.10: Panda-D Discovery Rate Approximation for the $N = 5$ experiment from Fig. 3.12: Error rate (%) of the experimental per-link discovery rate compared to (3.20).

	1	2	3	4	5
1	-	-17	8	-15	-9
2	-32	-	-22	-23	-20
3	12	12	-	-4	23
4	-17	-20	-17	-	12
5	-11	13	16	11	-

In the Birthday protocol, each node transmits a beacon in a slot with probability p . Therefore, the power budget is simply

$$p(2P_t M + P_r(d_s - 2M) + C_{st} + C_{ts}) \leq P_b.$$

In our evaluation, we select t and p such that the power budget is fully consumed and term these protocols, Searchlight-E and BD-E.

We note that there are numerous aspects of related protocols [24, 53, 88, 115, 149] which have not been considered. Specifically, existing works do not consider collisions occurring due to no clear channel assessment.⁹ Furthermore, numerous practical parameters are not considered such as the setting of the slot size. As can be seen above, the slot size impacts the average power consumption. In our simulation of Searchlight-E and BD-E, we ignore collisions and set the slot size to $d_s = 50\text{ms}$ with a guard time of 1ms, as was done in [149].

⁹ Panda avoids collisions by always listening before transmitting.

3.9.7 Approximate Analysis of Panda-D

We now approximate the directional discovery rate U_{ij} (i.e., the rate at which node i is discovered by node j) for a link of size $N = 2$, under non-homogeneous power budgets. Using similar analysis as described in Section 3.4, we obtain

$$U_{ij} = \frac{\frac{\lambda_i}{\lambda_i + \lambda_j}(1 - e^{-\lambda_j l})}{\frac{1}{\lambda_i + \lambda_j} + l + M}, \quad (3.20)$$

in which $\frac{\lambda_i}{\lambda_i + \lambda_j}$ is the probability that node i becomes the transmitter in a renewal. To evaluate this approximation, we apply it to the non-homogenous power harvesting experiment presented in Fig. 3.12.

Recall that the sleep rate for a node i , λ_i , is dynamically changing in Panda-D. We estimate the sleep rate based on the experimental average capacitor voltage using (3.17), which is shown in Table 3.9.

Corresponding to the parameters in Table 3.9, in Table 3.10 we compute the error rate between the experiment per-link discovery rate and (3.20). In general, the approximation is quite crude (typically within 25%). Yet it can still be used to as a rough approximation of the per-link discovery rate.

We remark here that the relatively high errors come from: (1) the small number of discoveries, (2) the fact that each node is performing independently without knowing the the value of N as a-priori, and (3) errors in the ADC capacitor voltage sampling. We observe that even after relaxing assumptions (A1, A2, A3), Panda-D still has robust performance in terms of the per-link discovery rate.

Chapter 4

Exploiting Mobility in Proportional Fair Cellular Scheduling: Measurements and Algorithms

Proportional Fair (PF) scheduling algorithms are the de-facto standard in cellular networks. They exploit the users' channel state diversity (induced by fast-fading), and are optimal for stationary channel state distributions and an infinite time-horizon. However, *mobile users* experience a non-stationary channel, due to *slow-fading* (on the order of seconds), and are associated with basestations for short periods. Hence, in this chapter, *we develop the Predictive Finite-horizon PF Scheduling ((PF)²S) Framework that exploits mobility*. We present *extensive channel measurement results* from a 3G network and characterize mobility-induced channel state trends. We show that a user's channel state is highly reproducible and leverage that to develop a *data rate prediction* mechanism. We then present a few *channel allocation estimation algorithms* that exploit the prediction mechanism. Our *trace-based simulations* consider instances of the (PF)²S Framework composed of combinations of prediction and channel allocation estimation algorithms. They indicate that the framework can increase the throughput by 15%–55% compared to traditional PF schedulers, while improving fairness.

This work is based on a collaboration with AT&T Labs-Research and has appeared previously

in [113, 114] as well as a pending patent application [13].

4.1 Introduction

3G and 4G (LTE) cellular networks are an enabler of intelligent transportation systems. They incorporate opportunistic schedulers [32] and allocate resources to users with good channel conditions by leveraging channel state variations, due to fast-fading,¹ as well as multi-user diversity. *Proportional Fair* (PF) scheduling algorithms are the de-facto standard for opportunistic schedulers in cellular networks [77]. They aim to provide high throughput while maintaining fairness among the users. PF scheduling algorithms have been extensively studied in the past (e.g., [18, 31, 91]). These algorithms are optimal under the assumptions that the wireless channel state is a stationary process (i.e., it is subject only to fast-fading) and the users' association times are long (e.g., static users or pedestrians) [96, 146]. However, when these assumptions do not hold (which is the case for mobile users), the performance of these algorithms is suboptimal [17].

For example, Fig. 1.3 illustrates a trajectory of a car along a 5km path, and the signal quality (E_c/I_o) to 3 different sectors (we collected the E_c/I_o values during 3 drives on the path). As can be seen, the channel has a dominant *slow-fading* component² on which the fast-fading component is overlaid. Since E_c/I_o has noticeable trends over several seconds, the channel state distribution is non-stationary. Additionally, movement along the path initiates hand-offs between the sectors, and therefore, the association periods are short. Since PF schedulers are not optimized for mobility, we will design a scheduling framework which can account for predictable mobility.

In Chapter 4, we design the Predictive Finite-horizon PF Scheduling ((PF)²S) Framework *which is tailored for mobile nodes and that takes advantage of both slow- and fast-fading*. It includes three components: (i) data rate prediction, (ii) estimation of future channel allocations,

¹Fast-fading is characterized by rapid fluctuations in the received signal strength (due mainly to multipath) [134].

²Slow-fading is characterized by slow (on the order of seconds) changes of the received signal strength (e.g., due to path loss and shadowing) [134].

and (iii) slow-fading aware scheduling.

To characterize slow-fading, to provide input to the design of the rate prediction mechanism, and to obtain traces for the evaluation of the framework and algorithms, we conducted an extensive measurement campaign. In particular, we discuss fine-grained (i.e., millisecond resolution) measurements, collected from a 3G network.³ Specifically, we measured wireless channel attributes in *drives spanning 810km and during a period of over 1,300 minutes*. Unlike a few previous studies (e.g., [141]) which measured the Received Signal Strength Indicator (RSSI) which is the total received power in a frequency band, *we measured the signal quality to each sector (E_c/I_o)*. This allows us to obtain important insights, since E_c/I_o is the most relevant predictor of a user's data rate.

We analyze the traces and show that mobile users experience pronounced slow-fading. However, the slow-fading trends cannot be simply tied to line-of-sight metrics, and therefore, developing simple channel state predictions is infeasible. Yet, the slow-fading component of E_c/I_o is remarkably reproducible for multiple drives on the same path (e.g., Fig. 1.3(b)), lending itself to data-driven prediction approaches.

Based on these observations, we develop a 2-phase rate prediction mechanism (referred to as the Coverage Map Prediction Mechanism (CMPM)). In an offline phase, measurement traces are processed to construct channel quality maps. The online phase is conducted by the sector and includes determination of the user's location and velocity, and thereby the predicted data rate. The localization can be simply done by querying the user's GPS. However, since this imposes energy and computation burdens on the user, we also develop the Channel History Localization Scheme (CHLS) which requires some knowledge of the user's trajectory.⁴ CHLS uses a variation of the Dynamic Time Warping (DTW) algorithm (originally developed for speech recognition [130]).

The (PF)²S Framework also requires algorithms that estimate the future channel allocations based on the rate predictions. We propose three such heuristic algorithms with different degrees

³The measurements were collected from a 3G network, due to lack of ubiquity of LTE networks. Yet, our observations regarding slow-fading apply to 4G networks, as they operate at similar time-scales.

⁴As such, it is highly applicable to users on highways and major roads.

of robustness to prediction errors and different performance levels for relatively accurate predictions. Using test cases generated from the collected traces, we perform an extensive simulation evaluation of the (PF)²S Framework. We consider 9 framework instances, representing combinations of rate prediction mechanisms and channel allocation estimation algorithms. We show that various instances of the framework consistently outperform the PF scheduler. Specifically, throughput improvements in realistic mobile scenarios range from 15% to 55% (with maintained or improved fairness levels). Finally, we study the sensitivity of the framework and algorithms to various network parameters and assumptions, including number of users and delay constraints.

The main contributions of this work are 3-fold: (i) it demonstrates, based on an extensive measurement campaign, that mobile users experience a reproducible but non-stationary slow-fading channel; (ii) it provides a cellular scheduling framework (and corresponding algorithms), tailored for mobile users; and (iii) it shows (using trace-based simulations) that the framework can significantly improve performance.

The chapter is organized as follows. Section 4.2 discusses related work and Section 4.3 reviews channel state metrics and formulates the problem. Section 4.3.3 presents the scheduling framework. Section 4.4 discusses the measurements and characterization of slow-fading. A rate prediction mechanism is presented in Section 4.5 and algorithms to estimate future channel allocations are presented in Section 4.6. The framework and algorithms are evaluated in Section 4.7. We conclude and discuss future work in Section 4.8.

4.2 Related Work

Opportunistic Scheduling: As mentioned, opportunistic and PF scheduling have been extensively studied (e.g., [18, 32, 91, 96, 146]). PF scheduling algorithms using *fast-fading* channel state predictions appear in [26, 75] (without a prediction mechanism). Scheduling for *mobile users* is considered in [15, 31, 141], where the underlying assumption in [15, 31] is that the user's mobility patterns induce a stationary (and known) *slow-fading* channel. The algorithm of [141] schedules *a single user* using an RSSI-based prediction method *at time scales on the order of minutes*. On the other hand, we solve a *multi-user* scheduling problem at *finer time scales (tens*

of seconds) using an E_c/I_o -based prediction mechanism.

Channel Measurements and Predictions: Wireless channel measurement studies have been conducted for decades [16, 76]. Recently, [104] studied the interaction of applications and the physical layer attributes in the 1x-EVDO network (using a predecessor to our measurement tool). Slow-fading is studied in *controlled environments* in [147]. Methods for *short-term* (over a few milliseconds) prediction of non-stationary wireless channel states appear in [105]. The measurements in [170, 171] focus on the repeatability of achieved *bandwidth* in a 3G network. Unlike previous works, we conduct measurements of wireless channel quality in a 3G network to characterize and predict slow-fading patterns over tens of seconds.

Localization and Mobility Prediction: Localization in cellular networks includes approaches that utilize time of arrival, time-difference of arrival, angle-of-arrival, cell-ID, and received signal strength (see [80] and references therein). Mobility prediction schemes that utilize pattern tracking and learning algorithm are reviewed in [95]. The method in [33] uses the DTW algorithm, albeit for velocity estimation. The closest related works are [80, 141] that utilize RSSI in GSM networks to localize users via fingerprinting. On the other hand, our scheme uses multiple channel attributes (i.e., E_c/I_o and RSSI) as well as recent history and is evaluated via trace-based simulations.

4.3 Model and Problem Formulation

In this section, we review the channel state estimation process in 3G networks and formulate the scheduling problem.

4.3.1 Channel States in 3G Networks

In a 3G network [77], each basestation covers a cell which is divided into (typically 3) sectors. As illustrated in Fig. 1.3(a), for data scheduling and hand-off purposes, users estimate the wireless channel quality to each nearby sector. It is estimated as the ratio between the power of a sector-specific pilot signal and the total in-band power (including interference and noise), and is denoted by E_c/I_o . In Section 4.4, we will consider these values in our measurement study.

A user associates (connects) with the strongest neighboring sector, termed the *serving sector*, and is assigned a dedicated buffer at the sector. When the serving sector E_c/I_o value drops below a threshold (e.g., due to mobility), the user *hands-off* wherein it disassociates from the serving sector and connects to a new sector with a higher E_c/I_o value.

The downlink channel from the sector to the users is time-slotted. We will denote by $E_c/I_o[j]$ the value in time slot j . The users periodically report their E_c/I_o to the sector. Then, an appropriate channelization code is selected and mapped to a feasible *data rate*.⁵ The feasible data rate of user i in slot j is denoted r_{ij} . An *opportunistic* scheduler implemented in the sector utilizes the multiuser diversity of the data rates to allocate downlink slots to users (see Fig. 1.3(a)).⁶

4.3.2 Scheduling Problem Formulation

The common 3G scheduler solves a *Proportional Fair* (PF) Scheduling Problem [77,91] and aims to achieve high overall throughput while maintaining fairness among the users. The common assumptions regarding stationary channels and long association times do not hold in mobile scenarios (as will be shown in Section 4.4). Hence, we formulate the downlink scheduling problem as a variant of the PF Scheduling Problem while utilizing a formulation similar to [17] (which studied adversarial channels). Unlike previous work, (e.g., [31,96,146]), we *do not make assumptions regarding the channel state distributions* and optimize over a *finite time horizon*.

We assume that a sector has K associated users with backlogged downlink buffers.⁷ Denote by α_{ij} the scheduler allocation ($\alpha_{ij} = 1$, if user i is allocated slot j , and $\alpha_{ij} = 0$, otherwise). We denote the feasible data rate and the scheduler allocation matrices by $\mathbf{R} = \{r_{ij}\}_{K \times T}$ and $\boldsymbol{\alpha} = \{\alpha_{ij}\}_{K \times T}$, respectively. The nomenclature can be found in Table 4.1. We assume a finite

⁵The mapping from E_c/I_o to data rates is described in Appendix 4.9.1. The mapping is phone specific and for our phones, the maximum data rate is 20Mbps.

⁶Multiple users (typically, no more than 4) may share a slot. Practically, it is uncommon, and we assume that *exactly* one user is allocated a slot.

⁷While in practice the number of associated users varies with time, we focus on a specific time-period with a given number of users.

$E_c/I_o[j]$	The pilot SINR in time slot j
T	Duration of the time horizon (in time slots)
\tilde{T}	Duration of the time horizon (in seconds)
K	Number of users
r_{ij}	Feasible data rate for user i in time slot j
$\mathbf{R} = \{r_{ij}\}_{K \times T}$	Feasible data rates matrix
$\hat{\mathbf{R}} = \{\hat{r}_{ij}\}_{K \times T}$	<i>Predicted</i> feasible data rate matrix
α_{ij}	Fraction of time slot j allocated to user i
$\alpha = \{\alpha_{ij}\}_{K \times T}$	Allocation matrix
$\hat{\alpha} = \{\hat{\alpha}_{ij}\}_{K \times T}$	<i>Estimated</i> allocation matrix
d_i	User i 's accumulated delay since last service (number of time slots)
D_{starved}	Delay threshold at which a user is considered <i>starved</i>

Table 4.1: Nomenclature

time horizon of T slots that corresponds to the users' association times. By the end of slot T , user i accrues a *cumulative* service $\sum_{j=1}^T \alpha_{ij} r_{ij}$. Hence, we formulate the following problem where the objective is to maximize a *proportional fair* cost function.⁸

Finite-horizon Proportional Fair (FPF) Scheduling:

$$\max_{\alpha} \quad C = \sum_{i=1}^K \log\left(\sum_{j=1}^T \alpha_{ij} r_{ij}\right) \quad (4.1)$$

$$\text{subject to} \quad \sum_{i=1}^K \alpha_{ij} = 1 \quad \forall j = 1 \dots T \quad (4.2)$$

$$\alpha_{ij} \in \{0, 1\}. \quad (4.3)$$

Even with full knowledge of \mathbf{R} , this problem is NP-hard (the proof is provided in Appendix 4.9.2). In practice, this problem has to be solved in an online (causal) manner. Users are scheduled slot-by-slot, based only on *knowledge* of the history and without full knowledge of \mathbf{R} . While the objective in the FPF Scheduling Problem is to maximize the proportional fairness metric (4.1), when evaluating the framework (Section 4.7), we also consider the following metrics.

⁸Although we focus on proportional fairness, the general approach can be applied to other concave cost functions (e.g., the α -fairness class).

Definition 3 (Throughput). *The average data rate allocated to all users, $\sum_{i=1}^K \sum_{j=1}^T \alpha_{ij} r_{ij} / T$ is referred to as throughput.*

Definition 4 (Delay). *The number of consecutive time slots in which a user i does not receive an allocation is referred to as the delay and is denoted d_i . User i is starved if $d_i \geq D_{\text{starved}}$, where D_{starved} is a delay threshold.*

We note that in Section 4.7, the time horizon is sometimes considered in seconds, and is denoted by \tilde{T} .⁹

4.3.3 Predictive FPF Scheduling (PF)²S Framework

In this section, we review the widely deployed PF scheduling algorithm and present an online scheduling framework for solving the FPF Scheduling problem which combines two components: (i) data rate predictions and (ii) an estimation of future channel allocations. The design of these components will be presented in Sections 4.5 and 4.6, respectively. We first describe the PF scheduler deployed in 3G networks [77] which is used in later sections as a benchmark.

Definition 5 (PF-EXP [96, 146]). *The scheduler which sets $\alpha_{i^*j} = 1$ where*

$$i^* = \arg \max_{i \in K} r_{ij} / R_i[j],$$

and $R_i[j] = (1 - \epsilon)R_i[j - 1] + \epsilon\alpha_{ij}r_{ij}$, is referred to as PF-EXP.

In the definition of the PF-EXP scheduler, ϵ determines the tradeoff between throughput and delay. With large values of ϵ (≈ 1), the scheduler puts more weight on the users' current feasible rates, thereby improving throughput at the expense of delay performance. With small values of ϵ (≈ 0) the users allocation history has more weight, and therefore, the delay performance improves at the expense of throughput. The PF-EXP scheduler approaches *optimal* proportional fairness [96, 146] when the wireless channel state is a stationary process *and* users have long association times (i.e., $T \rightarrow \infty$).

⁹In HSDPA, which is the 3G technology used in our measurement campaign, the slot length is 2ms and hence, $\tilde{T} = T \cdot 2\text{ms}$.

Predictive FPF Scheduling (PF)²S Framework

- 1: Predict future data rates $\hat{\mathbf{R}} = \{\hat{r}_{ij}\}_{K \times T}$.
 - 2: Estimate future allocations $\hat{\alpha} = \{\hat{\alpha}_{ij}\}_{K \times T}$.
 - 3: **for** slot $j = 1$ to T **do**
 - 4: Compute $M_{ij} = \frac{r_{ij}}{\sum_{t=1}^{j-1} \alpha_{it} r_{it} + \hat{\alpha}_{ij} r_{ij} + \sum_{t=j+1}^T \hat{\alpha}_{it} \hat{r}_{it}} \forall i \in K$
 - 5: **if** $\exists i \in K$ with $d_i \geq D_{\text{starved}}$ **then**
 - 6: $i^* = \arg \max_{\{i \in K: d_i \geq D_{\text{starved}}\}} M_{ij}$
 - 7: **else** $i^* = \arg \max_{i \in K} M_{ij}$
 - 8: $\alpha_{i^*,j} = 1, \alpha_{i,j} = 0 \quad \forall i \neq i^*$
-

Our Predictive FPF Scheduling (PF)²S Framework follows a similar approach as the PF-EXP scheduler to make slot-by-slot allocations. It utilizes a gradient ascent approach [29] to maximize the objective function (4.1). In each time slot, the channel is allocated to the user corresponding to the largest objective function increase. Temporarily relaxing the integer constraints in (4.3), the gradient for user i in time slot j is:

$$\frac{\partial C}{\partial \alpha_{ij}} = \frac{r_{ij}}{\sum_{t=1}^T \alpha_{it} r_{it}} = \frac{r_{ij}}{\sum_{t=1}^{j-1} \alpha_{it} r_{it} + \alpha_{ij} r_{ij} + \sum_{t=j+1}^T \alpha_{it} r_{it}}. \quad (4.4)$$

Computing the above gradient *requires knowledge of the entire data rate matrix \mathbf{R} and is not feasible for an online algorithm*, which only has knowledge of the past. Hence, the denominator of (4.4) is broken up into three components (from left to right): *past*, *present*, and *future*. From the perspective of an online scheduler, the first two components are known in any time slot. To enable slot-by-slot scheduling, the *future* component of (4.4) is computed as part of the (PF)²S Framework, which is described in pseudo-code above.

Predictions of future data rates (r_{ij}) and estimates of future channel allocations (α_{ij}) are denoted by \hat{r}_{ij} and $\hat{\alpha}_{ij}$, respectively, with matrix representations denoted by $\hat{\mathbf{R}}$ and $\hat{\alpha}$. At time 0, predictions of $\hat{\mathbf{R}}$ and $\hat{\alpha}$ are *pre-computed* for the entire horizon (next T slots). These matrices can be generated using the methods described in Sections 4.5 and 4.6 but the framework can support other methods. For each user i in each slot j , a ranking M_{ij} which corresponds to (4.4), is computed using $\hat{\mathbf{R}}$ and $\hat{\alpha}$. The user with the highest ranking is selected.

For a stationary channel, the future channel statistics are captured in the past component of the denominator (4.4). Hence, algorithms that rely only on past information (i.e., PF-EXP) are optimal. However, for non-stationary channel distributions, this does not hold. Hence, unlike in PF-EXP, step 4 in the framework considers the future channel component. By incorporating the predicted future, the (PF)²S Framework can leverage *slow-fading* trends. In addition, by making slot-by-slot decisions, the framework also leverages *fast-fading* components, similar to PF-EXP.

Since the (PF)²S Framework aims to schedule users during slow-fading peaks (which may occur at several second intervals), it is essential to ensure that this does not result in long delays. Hence, in each slot, the framework first considers the set of *starved* users whose wait time d_i (from the last slot of service) exceeds D_{starved} (defined in Defn. 4) and selects one. If no user is starved, it selects among all users. Thereby, the framework can handle delay constraints. Note that delay considerations can be ignored by setting $D_{\text{starved}} = \infty$.

4.4 Slow-fading measurements

We now describe the measurements collected from a 3G network. Our analysis demonstrates that mobile users experience pronounced and reproducible slow-fading. The observation regarding reproducibility provides insights into the design of the data rate prediction ($\hat{\mathbf{R}}$) mechanism (Section 4.5).

4.4.1 Measurement Setup and Test Drives

The measurement campaign was conducted with Samsung Galaxy S II (GSII) Skyrocket phones [136] using the setup shown in Fig. 4.1. The phone was connected via USB to a laptop running the Qualcomm eXtensible Diagnostic Monitor (QXDM) software. QXDM queries the phone in real-time and captures various physical layer attributes (described below) as well as GPS reports of location and velocity. QXDM records these measurements every 20ms, capturing the fast-fading and slow-fading components.

For the mobile measurements, the setup was placed in a car which traversed 4 different

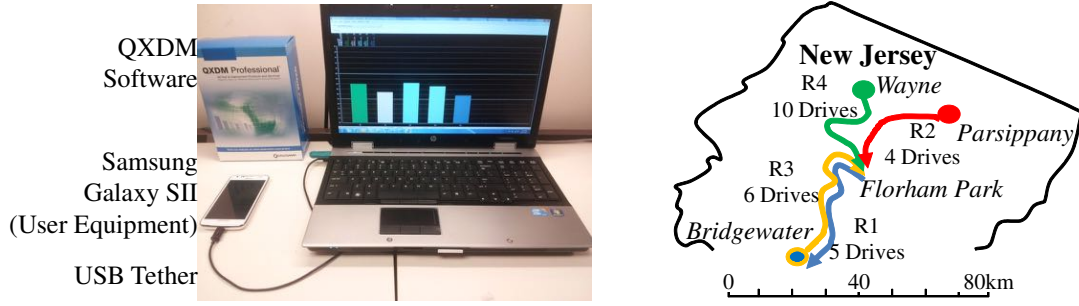


Figure 4.1: Measurement setup and the routes on which measurements were collected (see also Table. 4.2).

Label	Num Logs	Time Logged (min)	Total Dist. (km)	Av. Dist. (km)	Av. Velocity (m/s)	Total Sectors	Total Serving Sectors	Total Data (MB)
R1	7	305.4	205.5	29.4	6.6	282	67	246
R2	4	85.2	33.0	8.2	10.7	245	42	254
R3	6	252.4	220.8	36.8	21.9	210	68	251
R4	10	359.5	351.0	35.1	16.8	963	336	1538
Static	5	383.4	-	-	-	58	9	895
Total	32	1386.1	810.8	-	-	1758	522	3184

Table 4.2: Summary of collected measurements

routes¹⁰ that span both highways and suburban roads (see Table 4.2). Each drive followed the entirety of a given route and several drives were conducted. For control purposes, we also performed measurements with a *static* (immobile) setup. During the measurements, a continuous download was conducted to ensure a sustained network connection. In summary, we measured wireless channel attributes during drives spanning 810km and during a period of over 1,300 minutes.

¹⁰Note that routes R1 and R3 have the same start and end locations but are oriented in opposite directions. Hence, we treat them separately.

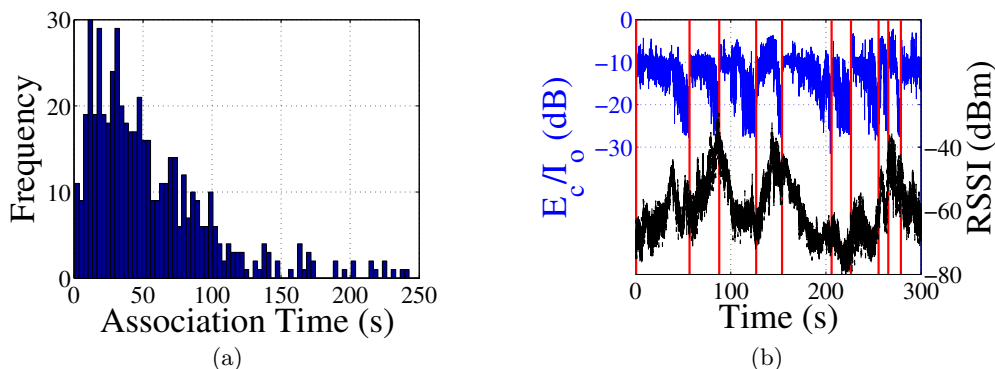


Figure 4.2: (a) Distribution of sector association times for 27 drives along routes R1–R4 and (b) measured values of the RSSI and the serving sector E_c/I_o for a drive on part of route R4 (vertical bars indicate hand-offs).

4.4.2 Channel State Metrics and Dynamics

QXDM stores three physical layer attributes: the total in-band power (including interference and noise), termed RSSI, the received pilot-power (RSCP), and the ratio between the pilot power and the total interference (E_c/I_o).¹¹ These key attributes characterize the channel quality and are periodically reported by the user to the serving sector [77]. While the latter two are *specific* to each nearby sector’s pilot channel, the former (RSSI) is not. Moreover, while RSSI was commonly logged and used in previous work (e.g., [141]), from a scheduling perspective, E_c/I_o is the most relevant indicator of a user’s channel quality [77].

We highlight the slow-fading phenomenon with an example. Fig. 4.2(a) shows a histogram of the users’ association times for 27 drives on all routes, demonstrating that association times are on the order of tens of seconds. As a specific example, Fig. 4.2(b) shows measured traces of the RSSI and the serving sector E_c/I_o for part of a single drive along route R4. Clearly, RSSI does not always reflect the same trend as E_c/I_o . Additionally, the E_c/I_o experiences slow-fading on the order of several seconds. Since in most cases, the user’s association times are tens of seconds, the slow-fading peaks and troughs occur *within each sector*. Therefore, *we focus in the next two subsections on E_c/I_o slow-fading trends*, which are leveraged by the (PF)²S Framework.

¹¹ $E_c/I_o(\text{dB}) = \text{RSCP}(\text{dB}) - \text{RSSI}(\text{dB})$.

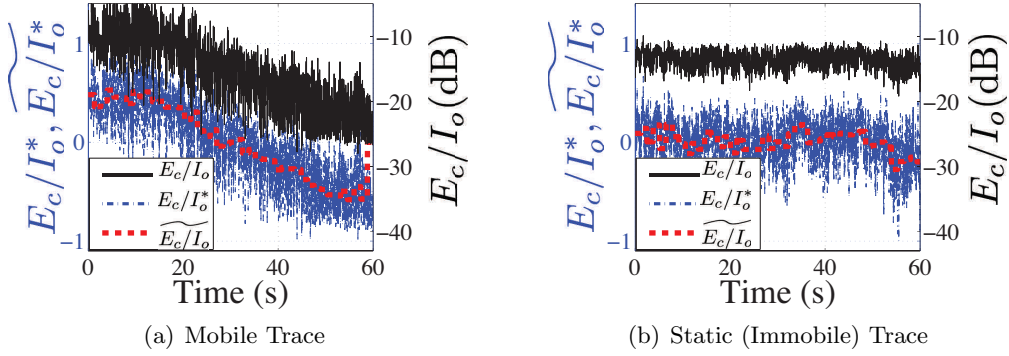


Figure 4.3: Comparison of E_c/I_o (measured), E_c/I_o^* (normalized), and $\widetilde{E_c/I_o^*}$ (smoothed): (a) a mobile trace from route R4 and (b) a static trace.

4.4.3 Slow-Fading and Mobility

We first demonstrate that the slow-fading phenomenon is closely tied to user mobility. We then characterize the correlation between slow-fading and mobility metrics and show (in contrast to assumptions in past work, e.g., [15]) that slow-fading trends cannot be tied to simple line-of-sight metrics.

To quantify the slow-fading in a user's E_c/I_o trace of T slots, we define a *slow-fading metric* as described below. First, the mean is removed and the trace is normalized to obtain:

$$E_c/I_o^*[j] = \frac{E_c/I_o[j] - \overline{E_c/I_o}}{\max_{1 \leq j \leq T} |E_c/I_o[j] - \overline{E_c/I_o}|}. \quad (4.5)$$

The operation does not affect E_c/I_o trends, but removes the amplitude which can vary depending on the sector, thereby enabling a comparison of E_c/I_o traces from different sectors. Then, E_c/I_o^* is *smoothed* by using wavelet transforms to remove the fast-fading components (with frequencies greater than 1Hz). The smoothed version of E_c/I_o^* is denoted by $\widetilde{E_c/I_o^*}$ (more details regarding the smoothing operation appear in Appendix 4.9.3). Fig. 4.3 provides visual examples of E_c/I_o^* and $\widetilde{E_c/I_o^*}$ for a mobile user and a static user. Using $\widetilde{E_c/I_o^*}$ clearly illustrates the presence (absence) of a trend in the values of E_c/I_o over the time-period of observation.

Finally, we define the *slow-fading metric* as $S(\widetilde{E_c/I_o^*}) = \sum_{j=1}^T \widetilde{E_c/I_o^*}[j]^2 / T$. E_c/I_o traces with no appreciable trends will have $\widetilde{E_c/I_o^*}$ values close to zero (Fig. 4.3(b)) and hence S will be small.

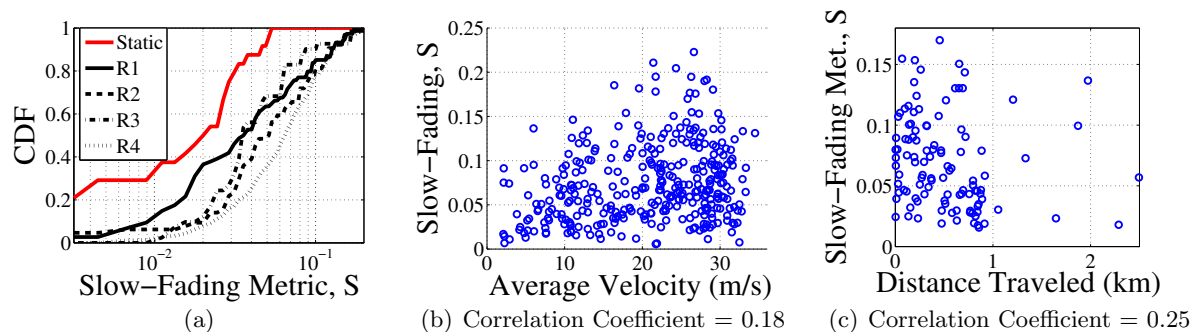


Figure 4.4: Characterization of slow-fading: (a) the CDF of the slow-fading metric (S) for mobile and static traces and (b)–(c) scatter plots for all mobile traces of the average velocity or distance traveled while connected to a sector vs. S .

On the other hand, E_c/I_o traces with noticeable trends (Fig. 4.3(a)) will have large values of $\widetilde{E_c/I_o^*}$ (positive or negative) and hence larger values of S . As it is normalized by T , S is used to compare E_c/I_o traces with varying association times from different sectors.

We computed S for every E_c/I_o trace collected from every serving sector (see Table 4.2). Fig. 4.4(a) shows the Cumulative Distribution Function (CDF) of the slow-fading metric values for routes R1–R4, as well as for the static traces. The mobile routes have much larger values of the slow-fading metric, confirming empirically that S accurately distinguishes mobile traces with slow-fading from immobile traces.

Previous work assumed a strong correlation between slow-fading and line-of-sight parameters (i.e., distance or velocity) [15, 32], supporting a functional prediction of the channel quality. However, our analysis indicates weak correlation between slow-fading (S) and line-of-sight metrics. For example, scatter plots of S and the average velocity or distance traveled of the user are shown in Figures 4.4(b) and 4.4(c), respectively. The correlation coefficient between S and the velocity or the distance is 0.18 and 0.25, respectively. Instead, slow-fading is governed by factors such as hand-offs, landscape, and movement-induced shadowing, which are complex to model even in controlled scenarios [147].

4.4.4 Slow-Fading Reproducibility

As described above, the slow-fading trend is not directly associated with line-of-sight factors, and therefore, simple functional predictions are infeasible. Yet, the slow-fading component of E_c/I_o is remarkably reproducible, enabling a data-driven prediction approach. Specifically, we observed that the E_c/I_o from multiple measurements (from separate drives) is predictable with an error of 1–3dB (a similar result appears in [141] for RSSI). To illustrate the reproducibility, we divide part of route R4 into 25m segments¹² and show in Fig. 1.3(b) the E_c/I_o observed across a subset of segments for 3 of the drives on the route through 3 sectors. The overlap of the curves indicates the similarity across all drives.

We strengthen this observation by computing the cross-correlation of E_c/I_o across *all* drives for each route, as follows. Each route is divided into 25m segments and each drive on this route is then represented by a *vector* of E_c/I_o values, one for each segment (e.g., if a route includes n segments, each drive is represented by a n -length vector, with multiple observations in the same segment represented by their average). We then compute the correlation coefficients of all the vectors (drives). Figures 4.5(b)(c)(d) show the correlation between all drives on routes R1, R3, and R4. Correlation coefficients are between 0.9–0.98, indicating a very high degree of correlation. The high correlation across all repeated drives implies that location-tagged historical measurements of E_c/I_o can be used to accurately predict future slow-fading.

4.5 Feasible Data Rate Prediction ($\hat{\mathbf{R}}$)

The (PF)²S Framework requires a mechanism to predict the users' feasible data rates for T slots ($\hat{\mathbf{R}}$). We design such a mechanism, based on the observation that the slow-fading component of E_c/I_o is highly reproducible, and refer to it as the **Coverage Map Prediction Mechanism (CMPM)**. In an offline phase, measurement traces are processed to construct geographic coverage maps. The online phase is conducted by the sector and is composed of two steps. First, the user's location and velocity are determined. Then, this information is used in conjunction

¹²25m is the minimum guaranteed GPS resolution.

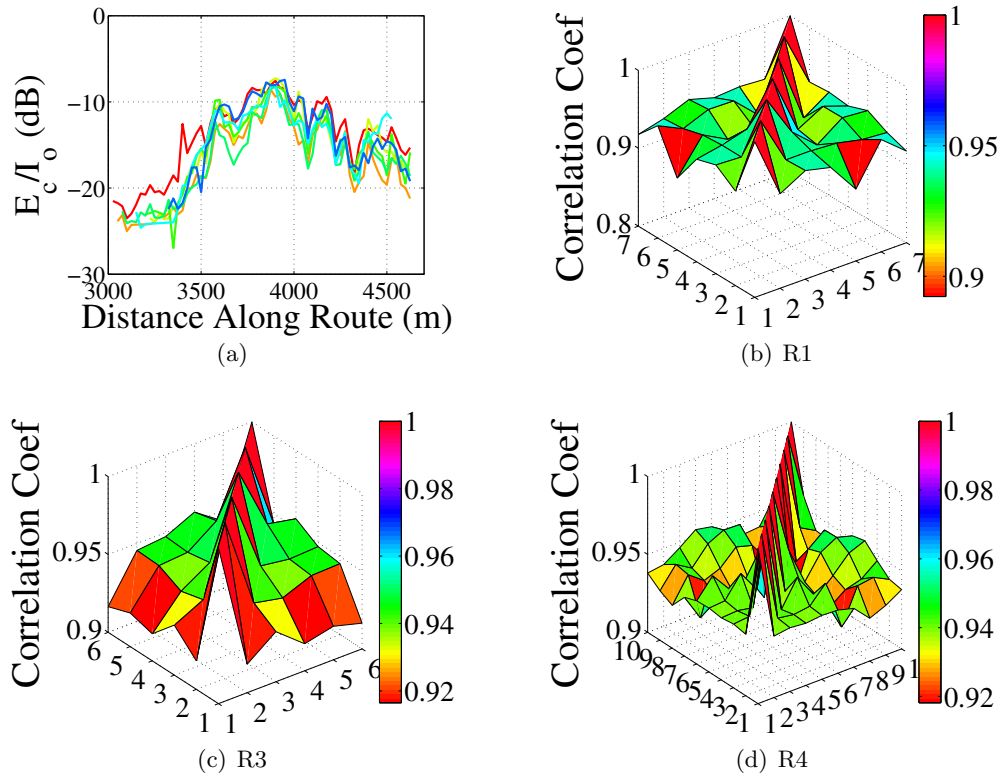


Figure 4.5: Reproducibility of measured E_c/I_o values: (a) measurements from 7 drives through a sector on route R1 (aligned in 25m segments) and (b)–(d) the correlation coefficient of E_c/I_o across all drives (with sufficient data) on routes R1, R3, and R4.

with the coverage map to predict user i 's feasible rates $\hat{r}_{ij} \forall 1 \leq j \leq T$.

The first step can be implemented by querying the user's GPS. However, since this imposes energy and computation burdens on the user, we also develop the Channel History Localization Scheme (CHLS). The scheme assumes that knowledge of the user's overall trajectory exists. In Section 4.7, we evaluate the framework using both alternatives.

4.5.1 Coverage Map Construction

The coverage map is constructed offline (once for each route) by placing a lattice over the geographic plane, and dividing it into square *segments* (see Fig. 4.6). Each segment, denoted by b , is covered by a set of sectors to which a user residing in it can associate, denoted by U_b .

Cellular carriers routinely measure the channel quality on major routes. These measurements can be used to compute, for each segment b , an *average* RSSI value as well as average values of E_c/I_o and RSCP for every nearby sector $u \in U_b$. These are denoted by $\overline{\text{RSSI}}\langle b \rangle$, $\overline{E_c/I_o}\langle b \rangle$, $\overline{\text{RSCP}_u}\langle b \rangle$. To compute these values for our evaluations, each sample measurement was tagged with a GPS location and tied to the appropriate segment.

4.5.2 Channel History Localization Scheme (CHLS)

The first step of the CMPM online phase localizes the user in the coverage map. To do this without GPS, we develop the CHLS. It matches the user's historical channel quality to coverage map segments on the user's trajectory, based on the differences between the channel metric values. Then, the user's location is estimated as the segment paired with its current channel quality value. Matching the channel quality history (i.e., a time-series) to segments (i.e., locations) depends on the user's velocity, which can vary. Hence, we utilize the Dynamic Time Warping (DTW) Algorithm¹³ to 'unwarp' the user's historical channel qualities to best fit the coverage map.

The CHLS requires knowledge of the user's trajectory and the user's location at a time slot in the recent history.¹⁴ The user's historical channel measurements are available at no extra cost as they are periodically reported to the network for scheduling and hand-off purposes.

The notation used to describe the scheme is defined below (see also Fig. 4.6). The sector keeps a history of the user's $E_c/I_{o_u}[j]$, $\text{RSCP}_u[j]$, and $\text{RSSI}[j]$ for the past T_p time slots, which are numbered sequentially from 1 to T_p (present slot). The coverage map segments are sequentially numbered, starting with the segment which is the user's estimated location at slot 1. The segments are numbered up to B_{\max} , which is the furthest segment in which the user could reside within the sector coverage area. The serving sector covers a range of segments $\mathbf{B} = \{b : B_{\text{cell}} \leq b \leq B_{\max}\}$.

¹³A similar dynamic programming algorithm is used in speech recognition [130] to align two phrases which are offset (in time, amplitude, etc.).

¹⁴For mobile users on highways and major roads, the trajectory can be estimated using mobility prediction techniques (e.g., [95]). The historical location can be reported based on past localization.

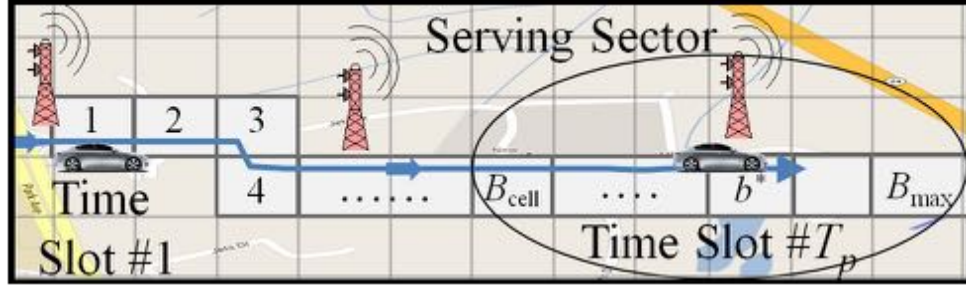


Figure 4.6: Illustration of the CHLS: the coverage map segments are labeled starting from 1 (at time slot 1, the user is in segment 1). At the present time slot (T_p), the user is located in one of the segments between B_{cell} and B_{max} , which fall within the coverage area of the serving sector.

The DTW Algorithm is applied to identify the cost of selecting each $b \in \mathbf{B}$ as the location estimate for the user. It constructs H , a matrix of size $B_{\text{max}} \times T_p$. The value of entry $h_{b,j}$ represents the minimum cost of pairing time slots from 1 to j with segments 1 to b . The constraint is that segment 1 is paired with slot 1 and segment b is paired with slot j (e.g., the end points are paired). The entries in the first row and column are, $h_{1,j}, h_{b,1} = \infty \forall b, j$, and the rest of the matrix is computed using $h_{b,j} = c(b, j) + \min(h_{b-1,j}, h_{b,j-1}, h_{b-1,j-1})$, where the cost of matching segment b to time slot j is

$$c(b, j) = (\text{RSSI}[j] - \overline{\text{RSSI}}\langle b \rangle)^2 + \sum_{u \in U_b} (E_c/I_{ou}[j] - \overline{E_c/I_{ou}}\langle b \rangle)^2 + (\text{RSCP}_u[j] - \overline{\text{RSCP}}_u\langle b \rangle)^2.$$

If channel quality history does not exist for $u \in U_b$ at slot j , then $c(b, j) = \infty$. Note that the CHLS uses all three channel quality attributes to increase accuracy. Moreover, for each time slot, it utilizes channel quality attributes corresponding to several sectors. The scheme concludes by estimating that the user resides in $b^* = \text{argmin}_{b \in \mathbf{B}} h_{b,T_p}$. To complete step one, the user's velocity is estimated, using training data to compute an average of past velocities near the estimated location.

The CHLS was evaluated via simulations. We set $T_p = 3,000$ slots (which corresponds to a horizon of 60s), set the segment size to $25\text{m} \times 25\text{m}$, and assumed that the serving sector coverage radius is 1,000m. We created coverage maps using half of the traces reported in Table 4.2.

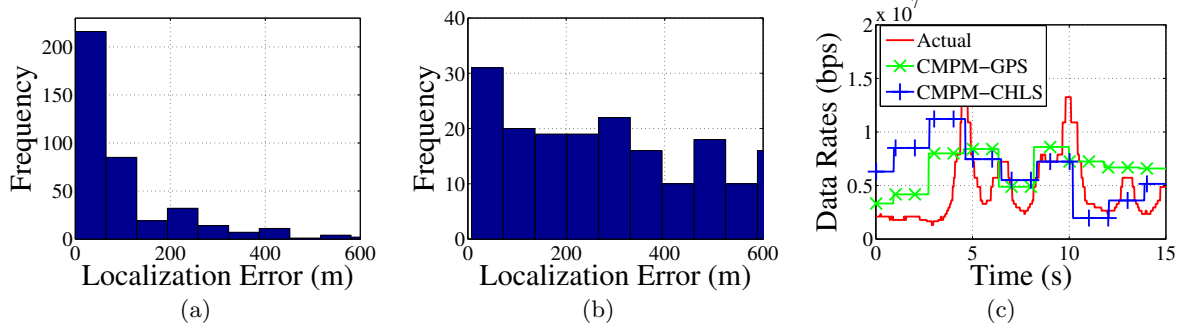


Figure 4.7: Evaluation of the Coverage Map Prediction Mechanism (CMPM): (a) CHLS error distribution for 500 tests compared to (b) the error distribution for 500 tests of the localization scheme provided in [141]; (c) An example CMPM data rate prediction when location and velocity are determined using GPS or CHLS.

From the remaining traces, we selected 500 random instances of 60s-length. The distribution of localization errors is shown in Fig. 4.7(a). The scheme has a median error of 23m and average error of 123m. For comparison, our evaluation of the RSSI-based localization scheme of [141] is shown in Fig. 4.7(b) and resulted in a median error over 300m.

4.5.3 Feasible Data Rate Prediction

Recall that the FPF Scheduling Problem formulation is based on feasible data rates. Hence, we now transition to using data rates. The relation between E_c/I_o and data rates (provided in Appendix 4.9.1) is monotonic, and therefore, the reproducibility conclusions from Section 4.4 also apply to data rates.

A simple online algorithm that operates in the sector estimates the user's future data rates using a coverage map and an estimate of the user's current location and velocity (either from GPS or the CHLS). First, future locations are predicted assuming that the velocity is constant for future time slots. Each location is then mapped to a segment in the coverage map which in turn yields a data rate.

Fig. 4.7(c) shows an example data rate prediction for the CMPM using the two variations, to which we refer as CMPM-GPS and CMPM-CHLS. In Section 4.7, we demonstrate that

the CPM-CHLS captures enough of the slow-fading effects when integrated into the (PF)²S Framework to improve scheduling performance.

4.6 Allocation Estimation ($\hat{\alpha}$)

The (PF)²S Framework (described in Section 4.3.3) requires a channel allocation ($\hat{\alpha}$) estimation algorithm based on the data rate predictions. This can be viewed as obtaining a solution to the FPF Scheduling Problem using the *predicted data rate matrix* $\hat{\mathbf{R}}$. As the framework operates in an online manner, the main design considerations are simplicity and robustness to prediction errors. We now introduce three algorithms which trade fairness and throughput performance for robustness to prediction errors. These algorithms will be evaluated with the rest of the framework in Section 4.7.

Round Robin Estimation (RRE): This simple heuristic assumes that future time slots are allocated in a *round-robin* manner and each user receives an *equal* number of slots, resulting in an estimated allocation of $\hat{\alpha}_{ij} = 1/K \forall i, j$.

Blind Gradient Estimation (BGE): This heuristic utilizes (4.4) to select a user in each slot, but *without* the future component (since it is not known). Specifically, starting from $j = 1$, it sets $\hat{\alpha}_{i^*j} = 1$ where $i^* = \operatorname{argmax}_{i \in K} (\hat{r}_{ij}) / \sum_{t=1}^j \hat{\alpha}_{it} \hat{r}_{it}$. The expression contains only slot indices $\leq j$, and is similar to PF-EXP. BGE requires $O(KT)$ operations.

Local Search Estimation (LSE): This greedy algorithm, described below, initiates with an objective function value C based on $\hat{\alpha}$ composed of random values. It iterates slot-by-slot, greedily allocating slot j to the user with the largest gradient value, assuming all other time slots are fixed. The algorithm proceeds cyclically (returning to slot 1 after T) until reaching a local-maxima (i.e., no change in T iterations). Termination is guaranteed as the objective value is bounded from above. Each cycle of LSE takes $O(KT)$ computations. Practically, it usually terminates after a few cycles.

An example of $\hat{\alpha}$ values obtained by each of the algorithms appears in Fig. 4.8(a). LSE's estimates are tightly clustered near the predicted slow-fading peaks. The estimates from BGE are more diffused and those of RRE are uniform. Consequently, if the rate predictions are

Local Search Estimation (LSE) Algorithm

Input: Predicted data rates $\hat{\mathbf{R}} = \{\hat{r}_{ij}\}_{K \times T}$.

Output: Estimated allocations $\hat{\alpha} = \{\hat{\alpha}_{ij}\}_{K \times T}$.

- 1: Choose an initial random $\hat{\alpha}$.
 - 2: $j = 1$, $LastChange = 1$, $C = \sum_{i=1}^K \log(\sum_{j=1}^T (\hat{r}_{ij} \hat{\alpha}_{ij}))$
 - 3: **repeat** $i^* = \arg \max_{i \in K} r_{ij} / \sum_{t \in \{1, T\} \setminus j} (\hat{r}_{it} \hat{\alpha}_{it})$
 - 4: $\hat{\alpha}_{i^*, j} = 1, \hat{\alpha}_{i, j} = 0 \quad \forall i \neq i^*$
 - 5: $C' = \sum_{i=1}^K \log(\sum_{j=1}^T (\hat{r}_{ij} \hat{\alpha}_{ij}))$
 - 6: **if** $(C' > C)$ **then** $C = C'$, $LastChange = j$
 - 7: $j = (j \bmod T) + 1$
 - 8: **until** $j \neq LastChange$
-

accurate, the framework using LSE provides the best performance, since it correctly allocates slots near the peak rates. The framework using BGE allocates slots around the peak rates, resulting in moderately good performance. The framework with RRE allocates slots uniformly, occasionally occurring during the peaks. On the other hand, if the prediction is erroneous, LSE would suffer, since it pushes the framework to schedule the user's slots at the predicted slow-fading peaks. BGE provides some robustness to prediction errors, and RRE is the most robust.

4.7 Performance Evaluation

We now use trace-based simulations to evaluate the performance of the (PF)²S Framework described in Section 4.3.3. Framework *instances* use combinations of CPM implementations and channel allocation estimation algorithms. The test cases are generated using measurement traces and the performance metrics are *proportional fairness* (4.1) and *throughput* (Defn. 3). We show that various instances of the (PF)²S framework consistently outperform the deployed scheduler (PF-EXP), with throughput improvements in realistic scenarios ranging from 15% to 55%, while maintaining similar delay performance. We then study the framework's sensitivity to the time horizon, number of users, mobility, $\hat{\mathbf{R}}$ accuracy, delay threshold, and coverage map

resolution.

4.7.1 Generation of Coverage Map and Test Cases

From the dataset presented in Section 4.4, half of the drives on each route were used as training measurements for coverage map construction (using a $25\text{m} \times 25\text{m}$ segment size). The remaining measurements were used for test case generation. A single test case was generated for every sector that had enough measurement data. Each test case is comprised of K users and T time slots and it emulates users starting at different locations within the sector coverage area and traveling with varying velocities in both directions along a route.

For each user i , the data rates r_{ij} , $1 \leq j \leq T$ were generated by selecting a segment of T random contiguous slots from part of the trace where the user was associated with the sector. In half of the cases, the vector was time-reversed, emulating travel in the opposite direction. Fig. 4.8(b) shows the data rates (r_{ij}) for an example test case with $K = 7$ and $\tilde{T} = 30\text{s}$ (recall that \tilde{T} is the time horizon in seconds).

Finally, for each generated rate matrix \mathbf{R} , we consider 3 approaches for obtaining the predicted rate matrix $\hat{\mathbf{R}}$: clairvoyant (a.k.a., complete knowledge, $\hat{\mathbf{R}} = \mathbf{R}$), the CPM which uses GPS information for location estimation (referred to as CPM-GPS), and the CPM which uses the CHLS (referred to as CPM-CHLS). Using these approaches enables evaluating the framework with different qualities of $\hat{\mathbf{R}}$ prediction.

4.7.2 Baseline Comparison and Upper Bound

The (PF)²S Framework is compared to the deployed scheduler, PF-EXP (see Defn. 5), by normalizing the throughput and fairness values by the corresponding values obtained by PF-EXP. Hence, metric values greater than 1 show improvements over PF-EXP. As an upperbound, the optimal solution to the FPF problem (referred to as OPT) is obtained using CVX, a MATLAB solver [1]. Note that OPT is obtained ignoring the integer constraints (4.3), using \mathbf{R} , and without delay constraints.

As mentioned in Section 4.3.3, the parameter ϵ *implicitly* controls the throughput-delay

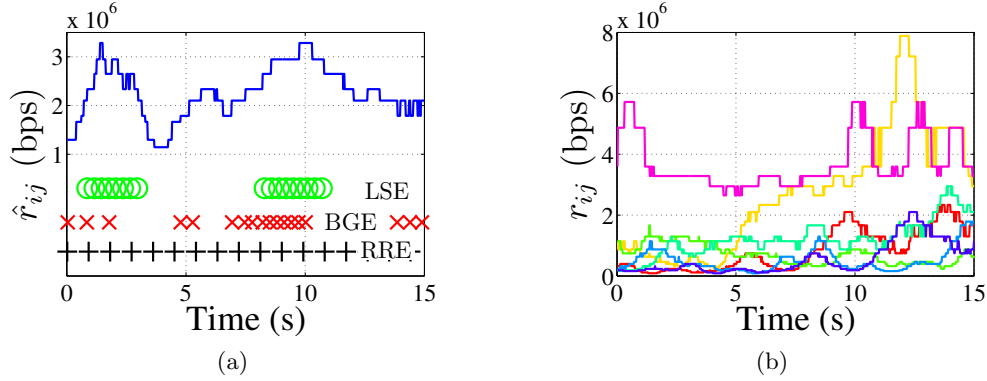


Figure 4.8: (a) An example of predicted data rates for a user with $\tilde{T} = 15$ s and the corresponding $\hat{\alpha}$ estimations computed by the LSE, BGE, and RRE algorithms. (b) The data rates (r_{ij}) for a test case with $K = 7$ and $\tilde{T} = 15$ s.

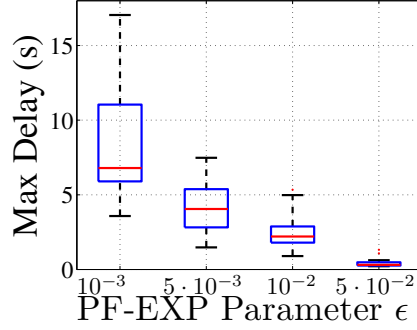


Figure 4.9: Statistical characterization of the maximum delay for the PF-EXP algorithm for 30 test cases with $K = 5$, $\tilde{T} = 30$ s, and varying ϵ .

tradeoff for the PF-EXP scheduler. The $(PF)^2S$ Framework *explicitly* controls the throughput-delay tradeoff using the parameter D_{starved} . Unless otherwise specified, we fix $D_{\text{starved}} = 0.5$ s. To ensure a fair comparison to our framework, we select a value of ϵ which results in the PF-EXP algorithm having similar delay performance. Correspondingly, based on box plots¹⁵ of the PF-EXP maximum delay shown in Fig. 4.9, we select $\epsilon = 0.01$. We discuss the sensitivity to these parameters in Section 4.7.4.

¹⁵Box plots include a whisker at maximum and minimum samples, a box at the 25th and 75th sample quantile, and a line at the sample median.

4.7.3 Throughput and Fairness Gains

We evaluate the throughput and fairness performance for various (PF)²S Framework instances and confirm experimentally that the $\hat{\alpha}$ estimation algorithms provide different degrees of robustness to rate prediction errors.

Figures 4.10(a) and 4.10(b) present box plots of the framework's fairness and throughput performance gains for 22 randomly generated test cases with $K = 7$ and $\tilde{T} = 30$ s (gains greater than 1 indicate improvements over PF-EXP). Since the objective function is logarithmic, the fairness gains are at the order of a few percent. The throughput gains over PF-EXP for all framework instances are significant (up to 70%). Clearly, the performance of a framework instance depends on the rate prediction accuracy ($\hat{\mathbf{R}}$) and the channel allocation estimation ($\hat{\alpha}$) algorithm. Hence, we consider framework instances, categorized by the $\hat{\mathbf{R}}$ prediction mechanism: **$\hat{\mathbf{R}}$ Clairvoyant:** The throughput gains are substantial (20% to 70%). As expected, based on the framework instance performance, the estimation algorithms are ranked by $\text{LSE} > \text{BGE} > \text{RRE}$. In general, the LSE performance with complete knowledge was near optimal.¹⁶

$\hat{\mathbf{R}}$ from CPM-GPS: Fig. 4.10(a) and 4.10(b) shows that the ranking between the $\hat{\alpha}$ estimation algorithms is $\text{BGE} > \text{RRE} > \text{LSE}$. As described in Section 4.6, BGE provides relative robustness to prediction errors, and hence with CPM-GPS it often outperforms LSE with throughput gains of 20% to 55%.

$\hat{\mathbf{R}}$ from CPM-CHLS: The instances using LSE and BGE show the largest performance decrease (compared to using complete knowledge). Yet, they still result in gains over PF-EXP. In general, we found that RRE is most resilient to errors and results in *significant* throughput gains of 15% to 50%.

To confirm that the gains of the (PF)²S Framework over the PF-EXP algorithm do not come at the expense of degraded user delay performance, Figures 4.10(c) and 4.10(d) illustrate the delay experienced by the users. Recall that the delay (see Defn. 4) refers to the duration in which a user receives no allocations. Fig. 4.10(c) shows the CDF of the delay experienced by users for

¹⁶For all test cases, when $D_{\text{starved}} = \infty$, the throughput when using LSE with complete knowledge is within 0.05% of the OPT throughput.

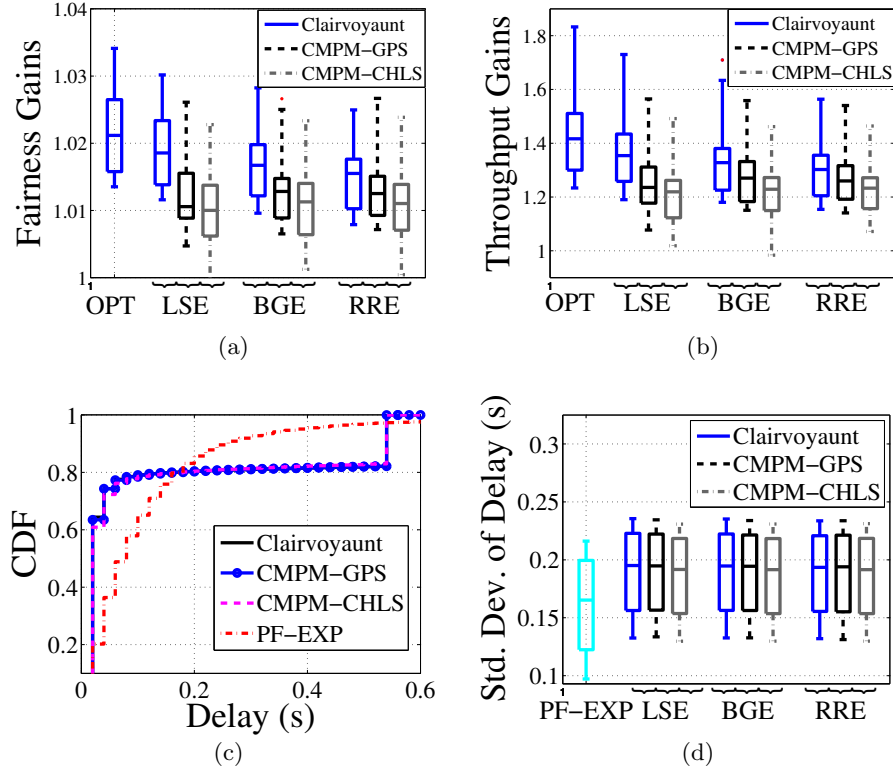


Figure 4.10: (PF)²S Framework performance for various framework instances (combinations of $\hat{\mathbf{R}}$ and $\hat{\alpha}$ prediction algorithms): statistical evaluation of 22 test cases with $K = 7$ and $\tilde{T} = 30$ s, and the resulting (a) fairness gains over PF-EXP, (b) throughput gains over PF-EXP, (c) cumulative distribution function of the users' delay for the BGE algorithm, and (d) standard deviation of the users' delay.

the (PF)²S Framework using BGE as well as the PF-EXP scheduler. Clearly, the delay performance of the framework is comparable to that of the PF-EXP scheduler, although the framework results in users having a slightly higher variation in the delay as indicated in Fig. 4.10(d). These results confirm that the (PF)²S Framework provides similar delay performance as the PF-EXP scheduler.

In summary, the evaluations with *real-world measurements* show that practical (PF)²S Framework instances match the PF-EXP scheduler in terms of delay performance, but consistently provide higher performance with throughput gains typically between 15% and 55%. The fairness gains, despite being logarithmic, are on the order of a few percent.

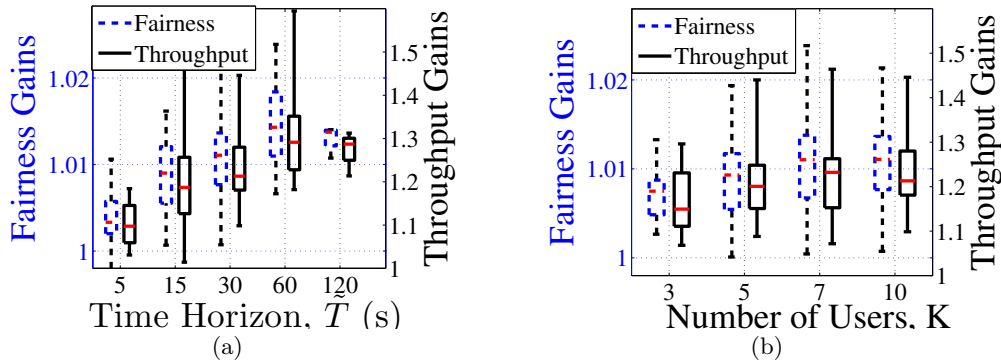


Figure 4.11: (PF)²S Framework performance gains when using RRE with CPM-CHLS: statistical evaluation of (a) 10 test cases with $K = 10$, varying the time horizon (\tilde{T}) and (b) 20 test cases of $\tilde{T} = 30$ s, varying the number of users (K).

4.7.4 Sensitivity Analysis

The results below are for the framework using RRE and CPM-CHLS (similar results for other framework instances provide equivalent conclusions and are therefore omitted).

Time Horizon (\tilde{T}): Fig. 4.11(a) shows the fairness and throughput gains for test cases with varying time-horizons \tilde{T} . Intuitively, larger \tilde{T} provides the framework additional opportunities to benefit over PF-EXP, which does not account for future data rates. For small to moderate values of \tilde{T} (5s, 15s), the framework shows 10%–30% throughput improvements. The performance gain for $\tilde{T} = 60$ s increases to 20%–60%. Eventually, as \tilde{T} grows, the framework becomes limited by the accuracy of the prediction, which decays with time.

Number of Users (K): Fig. 4.11(b) shows the fairness and throughput gains for 20 test cases with $\tilde{T} = 30$ s, and varying number of users. With additional mobile users, each experiencing their own slow-fading channel, multi-user diversity increases and the performance improves. The throughput gains increase from up to 25% with 3 users to up to 45% with 10 users.

Effect of Mobility: To ascertain the affect of static users, we evaluate test cases created from the mobile and the *static (immobile)* measurements. With static measurements, shown in Fig. 4.12(a), the wireless channel state distribution is stationary. Fig. 4.12(b) considers the framework performance for all algorithms (with complete knowledge, as predictions are irrelevant

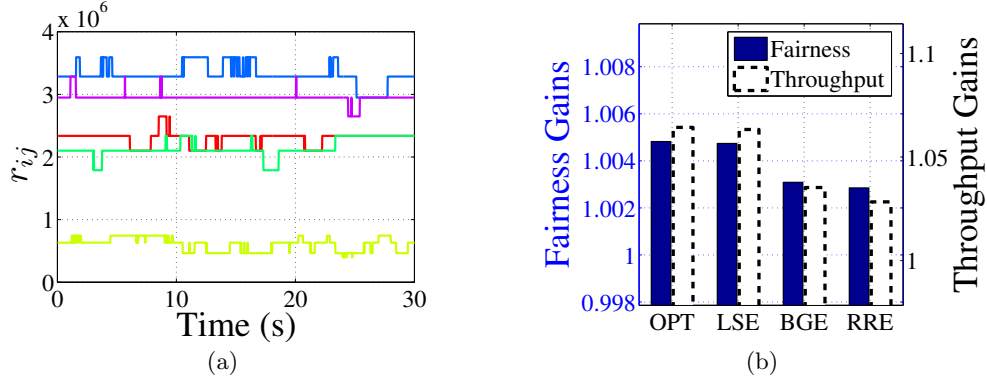


Figure 4.12: (PF)²S Framework evaluation of a test case generated using static (immobile) measurements with $\tilde{T} = 30$ s and $K = 5$: (a) the data rates r_{ij} and corresponding (b) fairness and throughput gains over PF-EXP using the BGE algorithm.

in this case) in a test case with $K = 5$ and $\tilde{T} = 30$ s. The framework performance is very similar to PF-EXP (with throughput gains within 6%) and is very close to OPT. Fig. 4.13 shows gains for 10 test cases of $\tilde{T} = 30$ s with 10 mobile users *and a varying number of static users*. With the addition of static users, PF-EXP performance improves (approaches optimal), and therefore, the gains decrease. Yet, due to the 10 mobile users, the gains are still quite *significant*, with throughput gains of over 30% in some cases.

Slow-fading Peak Prediction: As indicated, the accuracy of the $\hat{\mathbf{R}}$ prediction impacts the framework performance. Through careful inspection, we found that a key factor in prediction accuracy is the location of the slow-fading peaks. Hence, we now consider the impact of a simulated $\hat{\mathbf{R}}$ on the framework performance.

Specifically, the predicted rate vector for a subset of users is shifted by a certain amount, e.g., $\hat{r}_{ij} = r_{i(j-\text{offset})}$. Fig. 4.14 shows the fairness and throughput gains of the framework using RRE with subsets of the $K = 5$ users having an offset rate prediction. As expected, the performance decreases with large values of offset (both lead and lag). However, they are still quite significant ($\approx 20\%$ throughput gains). This experiment suggests that predicting the slow-fading peak within a few seconds of the actual peak will result in significant performance improvements. Additional tests confirm that this holds in more general scenarios.

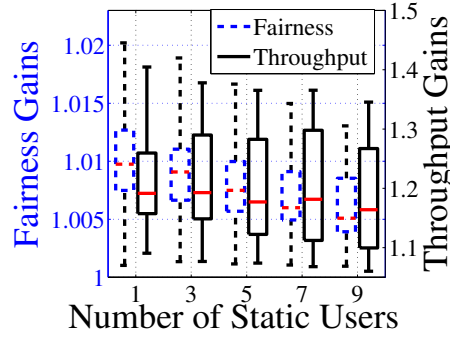


Figure 4.13: (PF)²S Framework performance gains when using RRE with CPM-CHLS: statistical evaluation of 10 test cases with $\tilde{T} = 30$ s, 10 mobile users, and varying number of static users.

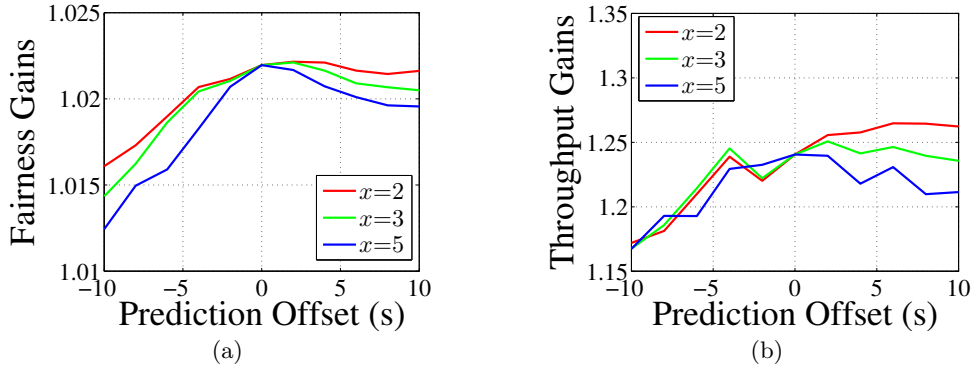


Figure 4.14: (PF)²S Framework evaluation for an example test case with $K = 5$ users, $\tilde{T} = 30$ s: (a) fairness and (b) throughput gains over PF-EXP using the BGE algorithm for varying the number of users, x , with simulated data rate prediction offsets.

Delay Parameters ($D_{\text{starved}}, \epsilon$): As described in Section 4.3.3, the framework uses the delay threshold D_{starved} to prioritize ‘starved’ users. The PF-EXP scheduler uses ϵ to implicitly control the delay. We now explore the performance gains achieved as we relax the delay parameters on both the PF-EXP scheduler (ϵ) and the (PF)²S Framework (D_{starved}). In Fig. 4.15(a), we vary these parameters (matching them based on Fig. 4.9). The figure shows, for comparable delay parameters over a range of delay tolerances (0.25–2s), that the (PF)²S Framework continues to outperform the PF-EXP scheduler, with the gains only mildly varying.

Coverage Map Resolution: In the results above, the coverage map segment size is $25\text{m} \times 25\text{m}$.

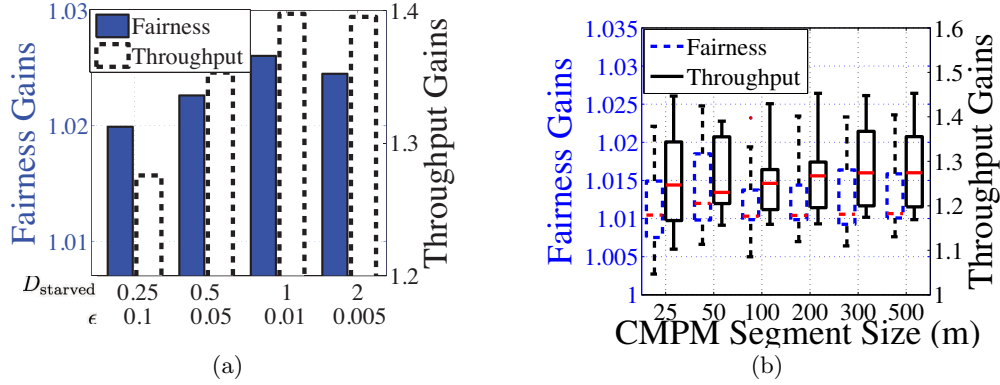


Figure 4.15: (PF)²S Framework performance gains when using RRE with the CPM-CHLS: (a) varying the delay parameters for the (PF)²S Framework (D_{starved}) and the PF-EXP scheduler (ϵ) for the test case given in Fig. 4.8(b), and (b) statistical evaluation of 15 test cases with $K = 10$ and $\tilde{T} = 30$ s for varying coverage map resolution values.

Fig. 4.15(b) shows the framework gains for 15 test cases with $K = 10$ and $\tilde{T} = 30$ s as a function of the map segment size. The performance does not degrade significantly as the segment size becomes reasonably large, since larger segments result in averaging of channel quality attributes over a larger area. This indicates that coarse channel measurements are useful for the framework.

4.8 Conclusions and Future Work

We described an extensive wireless measurement study as well as the design and trace-based performance evaluation of the (PF)²S Framework. We showed that by leveraging slow-fading, the framework (composed of various algorithms) can provide significant throughput gains while improving or maintaining fairness levels. Finally, we investigated the sensitivity of the results to different parameters and assumptions.

Future work will focus on relaxing some of the assumptions. Particularly, we plan to consider dynamic user populations handing-off between sectors. Additionally, we plan to extend the evaluations to consider policies that select appropriate $\hat{\alpha}$ estimation algorithms in different scenarios. Moreover, we will extend the localization scheme for cases in which trajectory information is unavailable or limited. Finally, as 4G networks become ubiquitous, we will conduct a

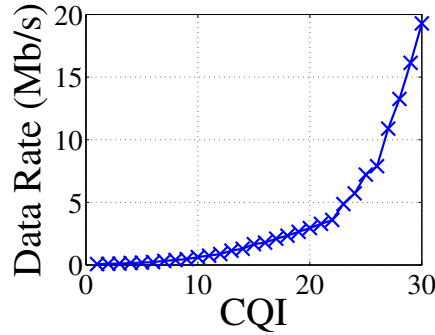


Figure 4.16: Mapping of Channel-Quality-Indicator (CQI) to feasible data rate for Galaxy S II phones, obtained from the 3GPP specifications [10].

corresponding measurement study to tailor the resource allocation algorithms.

4.9 Appendices

4.9.1 E_c/I_o to Data Rate Calculation

The Samsung Galaxy S II (GSII) phones [136] compute the E_c/I_o value for each serving sector. Based on empirical measurements, we mapped the E_c/I_o values to Channel-Quality-Indicator (CQI) values; measurements of E_c/I_o and CQI were taken simultaneously and, using a best-fit line, we computed a linear relationship between the two parameters. In conjunction with the phone's category, the CQI metric, which is an integer between 0-30, determines the available data rate to the phone. The GSII phones are category 14, and therefore, we used the mapping supplied in the 3GPP specifications [10] and shown in Fig. 4.16.

4.9.2 The FPF Scheduling Problem is NP-Hard

We show that the FPF Scheduling Problem is NP-hard even for the case of 2 users by a reduction from the Set Partition Problem, a classic NP-hard problem [59].

Theorem 2. *The FPF Scheduling Problem is NP-hard.*

Proof. Given a set of positive integers $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$, let $\mathcal{X}, \mathcal{A} \setminus \mathcal{X}$ be a partition of \mathcal{A} into

two disjoint sub-sets. Let $S_{\mathcal{B}} = \sum_{i \in \mathcal{B}} a_i$. The decision version of the Set Partition problem seeks to determine if there exists a set \mathcal{X} such that $S_{\mathcal{X}} = S_{\mathcal{A} \setminus \mathcal{X}}$. The decision version of the FPF Scheduling Problem asks if there exists an allocation such that the value of the objective function, (4.1), is at least C .

We construct a special case of the decision version of the FPF Scheduling Problem by setting $K = 2$ users and $C = 2\log(S_{\mathcal{A}}/2)$. Each user's channel has n slots with *identical* rates equal to \mathcal{A} . Specifically, in slot 1, both users have rate a_1 , in slot 2, users have rate a_2 and so on. The order of the rates is not critical, only that each user has identical rates in a slot, and as many rates as there are integers in \mathcal{A} .

We first show that the forward direction that a yes-instance of the Set Partition Problem results in a yes-instance of the FPF problem. Clearly, the existence of a set partition \mathcal{X}^* implies that $S_{\mathcal{X}^*} = S_{\mathcal{A} \setminus \mathcal{X}^*} = S_{\mathcal{A}}/2$. In the corresponding schedule from the FPF problem, let user 1 be allocated every slot corresponding to the items in \mathcal{X}^* and user 2 be allocated the remaining slots corresponding to the items in $\mathcal{A} \setminus \mathcal{X}^*$. The value of (4.1) in this scenario is $\log(S_{\mathcal{X}^*}) + \log(S_{\mathcal{A} \setminus \mathcal{X}^*}) = 2\log(S_{\mathcal{A}}/2)$. Therefore, this is a yes-instance of the FPF Problem.

Now consider the reverse direction. Assume that there does not exist a set partition. We argue by contradiction that this corresponds to a no-instance in the FPF Scheduling Problem. Assume there exists a schedule in the FPF problem with value $C = 2\log(S_{\mathcal{A}}/2)$. Let S_1 represent the sum of the rates allocated to user 1 and S_2 represent the sum of the rates allocated to user 2. As $S_1 + S_2 = S_{\mathcal{A}}$, it is clear that to achieve the optimal value of $C = 2\log(S_{\mathcal{A}}/2)$, $S_1 = S_2 = S_{\mathcal{A}}/2$. However, this is a contradiction of the non-existence of a set partition. Therefore, a no-instance of the Set Partition Problem corresponds to a no-instance of the FPF Problem. \square

4.9.3 E_c/I_o Smoothing - Review of Wavelet Transforms

The E_c/I_o signal is smoothed by applying a Discrete Wavelet Transform (DWT) which is ideally suited for finite, non-stationary signals. The approach follows the standard methodology (see [27] for an excellent description), which we briefly illustrate here. A DWT can be viewed as successive applications of a low-pass filter L and its mirror high-pass filter H , along with sampling to construct successively 'coarser' signal levels in terms of both time and frequency. For our

purposes, a Haar wavelet is used for the L and H filters.

Specifically, consider a signal x of length N sampled at rate f Hz. In an intermediate step, both the L and H filters are applied to obtain the coarse and high frequency signals, denoted by $L(x)$, $H(x)$, respectively. Each element of $L(x)$ and $H(x)$ is a wavelet coefficient and the two sequences are called the approximation and detail coefficients, respectively. $L(x)$ captures the ‘lower’ half of the frequency components, while $H(x)$ captures the ‘upper’ half of the frequency components. As each of the sequences are filtered to half of the bandwidth ($f/2$), they are sampled so that only $N/2$ samples are retained (as required by the Nyquist-Shannon sampling theorem).

Following this, the process continues by generating the sequence of wavelet vectors,

$$H(X), LH(x), L^2H(x), L^3H(x), \dots, L^{\log_2 N}H(x).$$

At level j , the wavelet $L^jH(x)$ has length of $N \times 2^{-j}$ and a bandwidth of $f/2^j$. Equivalently, since the samples in the original signal were sampled at rate f Hz, and hence are $\tau = 1/f$ seconds apart, the values at level j are $2^{j-1} \times \tau$ seconds apart. Hence, the signals at level j are ‘smoothed’, with variations at time-scales smaller than $2^{j-1} \times \tau$ removed.

The wavelet vectors across all the $\log_2 N$ levels form a complete orthonormal basis of the original signal.¹⁷ The measurements were collected at a time-interval of $\tau = 20$ milliseconds. To remove variations less than one-second, the wavelet vectors below level $\lceil \log_2 \frac{1}{\tau} \rceil \approx 6$ were removed. The remaining vectors $LH^7, \dots, LH^{\log_2 N}$ were recombined to obtain the smoothed signal.

¹⁷The original signal can be re-constructed by appropriately weighted combinations of the approximation and detail coefficients.

Chapter 5

Performance Evaluation of Fragmented Structures: A Theoretical Study

Dynamic spectrum access illustrate systems having a resource \mathcal{R} that allows fragmented allocations. We model \mathcal{R} as a sequence of $M > 1$ units for which requests are made by items in a FIFO queue; each request is for a desired amount of \mathcal{R} , the item size, and the residence time during which it is needed. So long as there are enough currently available units of \mathcal{R} , an item at the head of the queue can be divided into fragments accommodated by the gaps in \mathcal{R} formed by these units. Under the key assumption that allocations given to items can not be changed prior to their departures, fragmentation in the form of alternating gaps and allocated resource builds up randomly as items come and go. The improvements in resource utilization created by fragmentation are acquired at a processing cost, so how fragmentation evolves is an important performance issue.

We define a probability model specifying distributions for item sizes and residence times, and then analyze the system operating at capacity. If M is much larger than the maximum item size, then as the fragmentation process approaches equilibrium, nearly all of the allocated items are completely fragmented, i.e., the occupied units are mutually disjoint. In a suite of

four theorems, we show how this result specializes for certain classes of item-size distributions. However, as a counterpoint to these rather intimidating results, we present the findings of extensive experiments which show that the delays in reaching the inefficient states of nearly complete fragmentation can be surprisingly long, and hence that even moderately fragmented states are usually of no concern.

The results from this chapter were previously presented in [40].

5.1 Introduction

Traditionally, fragmentation in one-dimensional dynamic resource allocation has referred to segments of a resource assigned to items alternating with gaps of unassigned, potentially wasted resource. A classic example occurs in dynamic storage allocation [94]; the resource is computer memory and item requests are for the storage of files. Files must be stored in *consecutive* units/slots of memory. Depending on the application, these could be a sequence of bytes, words, or pages. Fragmentation is created solely by the random arrival and departure of files of varying sizes, and by the requirement that files not be moved once they are allocated. *The fragmentation of interest here is fundamentally more general*; it occurs when resources are such that, when no individual gap is sufficiently large, an item's allocation can be broken down into fragments that fit into multiple gaps. A small example will help fix ideas. In what follows, the notation u_i will often be used to denote both the name and size of the i -th item; context will always make clear which meaning applies.

In Fig. 5.1, the resource is 20 slots wide and the items, u_i , are for at most 10 slots. Successive departure times are denoted by the t_i . Allocations for the first 8 items u_1, \dots, u_8 are shown using a simple left-to-right first-fit rule. Assuming the resource is not in use at time 0, items u_1 , u_2 , and u_3 are the first to be allocated; u_4 must wait for a departure, since the first 4 item sizes sum to more than 20. The first occurrence of fragmentation occurs at the departure of u_2 and the resulting admission of u_4 ; an initial fragment of u_4 is placed in the gap left by u_2 and a final fragment is placed after u_3 . Also, even after u_2 and u_4 have departed, there is still not enough of the resource available for u_5 . After the additional departure of u_1 , both u_5 and u_6 ,

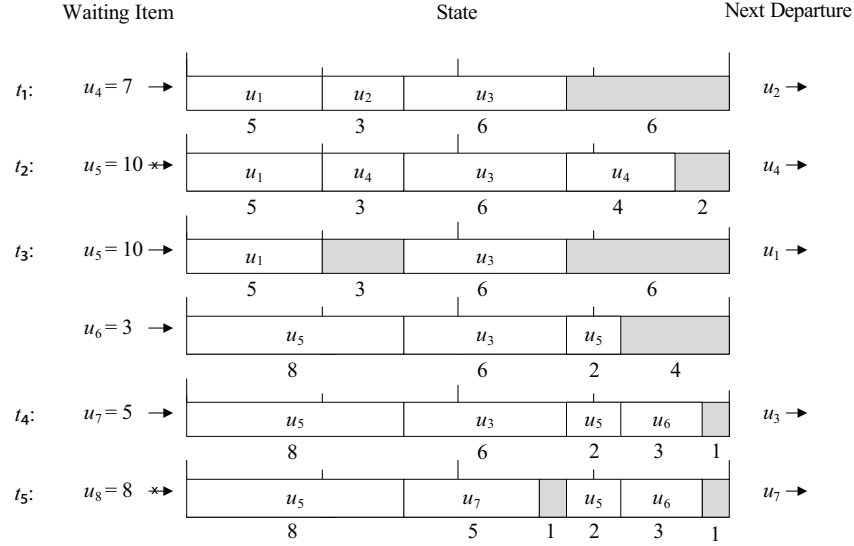


Figure 5.1: An example of admission and departure processes and the resulting fragmentation.

but not u_7 , can be allocated their requested numbers of slots.

As indicated above, the extent of fragmentation can have a significant impact on the efficiency of real systems and networks, and therefore, modeling and analysis can provide important insights. Intriguing problems of dynamic storage allocation have been around for over 40 years [39, 94], and they continue to resist a stochastic analysis leading to insightful results.

In Chapter 5, we study fragmentation within a probability model of the combinatorial structures illustrated in Fig. 5.1. Although our model does not refer to any particular application, it is applicable to many practical systems including cognitive radio as listed above. As such, we will use a general notation: a *resource* is composed of *slots* and represents, for example, the wireless spectrum, and an *item* represents, for example, a request for sub-channels.

Two intriguing questions are center stage: How bad can fragmentation become and at what rate does it develop? For example, can fragmentation progress to a point where nearly all items are *completely fragmented*, i.e., for every item size $j > 1$, are nearly all size- j items allocated j mutually disjoint slots? Answers to these questions give insights into those circumstances where defragmentation is worth the cost it incurs. We will discuss algorithmic issues, but only for illustrative purposes in our experimental study of convergence, since our main results on

stochastic fragmentation hold true for all allocation algorithms that one can expect to find in practice.

We define a probability model specifying distributions for item sizes and residence times, and then analyze the system operating at capacity. If the resource size is much larger than the maximum item size, then as the fragmentation process approaches equilibrium, nearly all of the allocated items are completely fragmented, i.e., the occupied units are mutually disjoint. In a suite of four theorems, we show how this result specializes for certain classes of item-size distributions. However, as a counterpoint to these rather intimidating results, we present the findings of extensive experiments which show that the delays in reaching the inefficient states of nearly complete fragmentation can be surprisingly long, and hence that even moderately fragmented states are usually of no concern.

Section 5.2 describes our stochastic model of fragmentation in detail beginning with a definition of the state space of the fragmentation process and the constraints of transition functions imposed on allocation algorithms. The section then introduces a general class of allocation algorithms, describing two in detail that are used in the remainder of the chapter when needed for experiments. The background in Section 5.3 briefly covers earlier results germane to our model. The results in Section 5.4, the centerpiece of the chapter, apply simultaneously to all algorithms in the class defined in Section 5.2. For given item-size distributions, the results of Section 5.4 describe the asymptotic fragmentation behavior in statistical equilibrium for large resource sizes. The most general of these results is a theorem showing that the fraction of partially fragmented items tends to 0 as the resource size tends to infinity. In the case where size-1 items have positive probability, the result is strengthened considerably: The expected number of partially fragmented items is shown to be bounded by a constant independent of the resource size. The case where there are only items of sizes 1 and 2 is treated separately, as a tight bound can be found for the constant in this case. Section 5.5 investigates the convergence times to highly fragmented states, relying primarily on simulation experiments. Section 5.6 concludes the chapter by suggesting areas that merit further research, with an analysis of algorithms being a prime example.

5.2 The Model

The first of the two subsections below introduces in broad terms the fragmentation process at the heart of the chapter; the class of allocation algorithms that completes the model definition is covered in the second subsection.

5.2.1 The Fragmentation Process

A queue of items waits to use a resource \mathcal{R} consisting of a sequence of M slots.¹ The resource is made available to items on a first-come-first-served (FCFS) basis, each item's request specifying the number of slots that it needs (i.e., the item's size), and the amount of time it needs the slots. The sizes are i.i.d. random variables governed by the distribution $\mathbf{q} = \{q_1, \dots, q_K\}$. To avoid trivialities we assume that at least two item sizes have positive probability, so that $q_K > 0$ for the maximum item size K . An item waits until it arrives at the head of the queue, at which point it is allocated its requested amount of \mathcal{R} as soon as the total size of the gaps in \mathcal{R} becomes large enough. The allocation of an item of size u , say, may be fragmented as desired to exploit gaps in resource usage, each of which is smaller than u , but whose cumulative size is at least u . Various algorithms other than the first-fit algorithm of Fig. 5.1 can be adopted to satisfy the requests of new items; we return to further discussion of allocation policies after a few more preliminaries.

The item at the head of the queue is usually called the HOL (head-of-the-line) item. In reference to allocations, the term *fragment* always refers to a piece of some item occupying a maximal sequence of consecutive slots, i.e., a slot adjacent to the sequence must either be empty or occupied by a different item.

We focus on the performance of systems operating at capacity, when throughput is at its maximum. Thus, the queue of waiting items never empties. Unless stated otherwise, the resource \mathcal{R} is initialized, as in the example of Fig. 5.1, with a sequence of unfragmented allocations to

¹A summary of the notation appears in 5.7.2.

items u_1, u_2, \dots such that

$$u_1 + \dots + u_{i-1} \leq M < u_1 + \dots + u_i.$$

At that point, all $i-1$ of the items begin independent, identically distributed residence times, or delays. Subsequent state transitions take place at departure epochs when the delays of currently allocated items expire. At such epochs, all fragments of \mathcal{R} in the allocation of a departing item are released, thus enlarging existing gaps and/or creating new ones. Suppose all items up to u_j have received their allocations and an item u_i , $i \leq j$, departs. Then items u_{j+1}, u_{j+2}, \dots are allocated to gaps until, once again, an item is encountered that requests more of \mathcal{R} than is available. All allocated items then begin or continue their residence times until the next departure. We emphasize that, once an allocation is made to an item, it remains fixed. Our probability model stipulates that item residence times are i.i.d. exponentially distributed random variables independent of item sizes. For convenience, the expected residence time is taken as the time unit. In summary then, for a given gap allocation policy, M and \mathbf{q} are the two parameters of the model.

More formally, a state x in the state space \mathcal{S} of a fragmentation process is given by

$$x = (L_1, \dots, L_N; u) \tag{5.1}$$

where u is the size of the item waiting at the head of the queue (the HOL item), $N \geq 1$ is the number of currently allocated items, and L_i , $1 \leq i \leq N$, is the list of disjoint subsequences of $(1, \dots, M)$ occupied by the fragments of L_i . For x to be admissible, the subsequences of L_i must be disjoint from those of L_j for all $i \neq j$, and for it to be *stable* u must exceed the resource available, i.e., $u > M - \sum_i \|L_i\|$, where $\|L_i\|$ is the size of the item (sum of the fragment sizes) in L_i . The fragmentation process is a Markov process for any allocation algorithm likely to be used in practice, and in particular, any algorithm in the class defined in the next section. Asymptotic analysis and the elementary theory of finite Markov chains provide the tools used in the analysis found in Section 5.3.

5.2.2 Allocation Algorithms

Our results characterize algorithms within a large class \mathcal{A} defined as follows. Let x be the state just after a departure and before new allocations are made and let A be the allocation algorithm. If x is not stable, then A computes an ordering of the gaps in x and allocates gaps to fragments of the HOL item u in that order. These gaps are allocated in their entirety to u , except possibly for the last gap. The fragment in the last gap is left-justified. If the new state x' is still not stable, allocations are repeated as above in state x' for the new HOL item. The process continues until a stable state is reached. An algorithm is said to have the *best-fit* property if the last gap to be allocated is the smallest remaining gap big enough to accommodate the last fragment.

The principal contribution of the chapter is the theory of the next section, which *applies to any allocation policy in \mathcal{A} —even one that computes a gap ordering based on the sizes and residence times of all future arrivals and departures.* However, we need algorithms for experiments targeting more precise behavior that we have been unable to analyze, a primary example being the convergence rates covered in Section 5.5. We introduce for illustrative purposes a fast scanning algorithm, Next-Fit (NF), and then a second, complementary algorithm, Best-Fit Decreasing (BFD), which is slower but which creates significantly less fragmentation. Both algorithms treat the resource as a homogeneous ring of M slots; slot 1 follows slot M and items and gaps seamlessly cross the boundary between these two slots. This assumption is largely for convenience and will have a negligible effect on results, since our primary interest is in cases where M is much larger than K .

NF is like the first-fit algorithm of Fig. 5.1 adapted to the ring, but it starts each scan where the preceding one left off. Specifically, a scan starts at the slot following the end of the last slot filled in making the preceding item allocation. NF is not a best-fit policy like BFD, which allocates an item to the gaps in decreasing order of size until the last fragment of the item (which could be the entire item) is smaller than at least one gap. At that point, the last, perhaps only, fragment of the item is allocated best-fit, i.e., to the smallest gap no smaller than the fragment. To be specific, we say that BFD breaks all ties in favor of the leftmost (lowest indexed) gap providing a best fit for the final fragment.

5.3 Background and Related Work

There are two processes closely related to our fragmentation process which have been studied in the past. Although the original model of item fragmentation was the one given here, the first results to appear were for the continuous limit of our model normalized on $[0,1]$, which was studied in [38]. In normalizing our model, the slot size is set to $1/M$ and the support of the item-size distribution is $\{1/M, \dots, K/M\}$. The continuous limit is reached as $K, M \rightarrow \infty$ subject to $K/M \rightarrow \alpha$, with $\alpha \in (0, 1]$ the single parameter of the model apart from the item-size distribution. The property of complete fragmentation in our model is analogous to behavior as $\alpha \rightarrow 0$ in the continuous model. Experiments have supported the claim that the number of fragments increases as $\alpha \rightarrow 0$ in the continuous limit, but this remains an open question. It is rather striking that the corresponding result is indeed provable in our original model, as shown in Section 5.4. The result gains in significance from experiments that have shown the continuous model to be distinctly limited in its approximation of the original, more realistic discrete scaling of this chapter. For example, under a uniform distribution on item sizes, one needs to assume that K is larger than about $M/10$ if the continuous limit with $\alpha = K/M$ is to give measures of fragmentation reasonably close to those of the model analyzed here. This is commonly an unacceptable assumption in practice.

The second related process is defined on the states $\hat{x} = (\|L_1\|, \dots, \|L_N\|; u)$ where fragment configurations are suppressed. The corresponding process $(X_A(t))$ remains Markovian for any algorithm $A \in \mathcal{A}$. Kipnis and Robert [92] generalized $(X_A(t))$ to a standard queueing process with stochastic arrivals. Their model was motivated by systems where fragmentation is not allowed; allocations are shifted as needed to keep available resource in one block.

The result to follow applies to an asymptotic, large- M analysis of the *gap-wait* process (G_t, H_t) at departure epochs, where G_t and H_t are respectively, at time t , the total gap size and the size of the HOL item. The result will clearly have no direct bearing on fragment configurations. Our focus is on behavior at departure epochs in statistical equilibrium, so we will usually drop the dependence on t . We include the result in this section, as it applies a key observation made in [92] that yields the asymptotics of the equilibrium probability distributions

for H and for G , both individually and jointly. We give our own proof as it does not require the considerably more extensive analysis of the general queueing model studied in [92].

Theorem 3. (*Kipnis and Robert [92]*) *At departure epochs, the following asymptotic equilibrium probability distributions apply. As $M \rightarrow \infty$, the distribution f_H for the HOL item size H tends to*

$$f_H(h) \sim \frac{hf_U(h)}{\mathbb{E}[U]}, \quad 1 \leq h \leq K, \quad (5.2)$$

the total gap-size distribution f_G tends to

$$f_G(g) \sim \frac{1 - F_U(g)}{\mathbb{E}[U]}, \quad 0 \leq g \leq K - 1, \quad (5.3)$$

and finally, the distribution π of the gap-wait state tends to

$$\pi_{gh} \sim \frac{f_U(h)}{\mathbb{E}[U]} \quad (5.4)$$

for all g, h satisfying $0 \leq g < h \leq K$. ■.

Examples: From (5.2), we have that

$$\mathbb{E}[H] \sim \frac{\mathbb{E}[U^2]}{\mathbb{E}[U]} \quad (5.5)$$

so with U uniformly distributed on $\{1, \dots, K\}$, calculations show that, as $M \rightarrow \infty$,

$$\mathbb{E}[H] \sim \frac{2K+1}{3}, \quad \mathbb{E}[G] \sim \frac{K-1}{3}, \quad (5.6)$$

and that the asymptotic gap-wait distribution is also uniform:

$$\pi_{gh} \sim \frac{1}{K(K+1)/2}, \quad 0 \leq g < h \leq K.$$

As another example, one to be used later, consider a general distribution for $K = 2$, with $q_1 := \Pr\{U = 1\}$. We find that $\mathbb{E}[U] \sim 2 - q_1$, and

$$\pi_{01} \sim \frac{q_1}{2 - q_1}, \quad \pi_{02} = \pi_{12} \sim \frac{1 - q_1}{2 - q_1} \quad (5.7)$$

as $M \rightarrow \infty$.

Proof of Theorem 1: As $M \rightarrow \infty$ for fixed K , the frequency of size- i items currently allocated in \mathcal{R} tends to $f_U(i)$ for all $i \in \{1, \dots, K\}$ by the laws of large numbers. Now let (U_i) be an equilibrium renewal process with intervals U_i between successive renewals having the item-size distribution F_U . Then the size of the interval covering the point M is, asymptotically for large M , equal in distribution to the size of the interval covering a specific point in the renewal process (U_i) . For convenience, we adopt the convention that, if M coincides with the end of one interval and the start of the next one, it is the latter interval that covers M . The key observation, extending an argument in [92], is that the duration of the interval covering M is equal in distribution to the size H of the HOL item and, asymptotically, is a random variable with the familiar distribution given in (5.2).

To calculate the corresponding, asymptotic total-gap size distribution, we note first that, in the asymptotic regime, for an HOL item size h , the total gap size has a uniform distribution on the set $\{0, \dots, h-1\}$, so

$$\begin{aligned} f_G(g) &\sim \sum_{g+1 \leq h \leq K} \Pr(G = g | H = h) f_H(h) \\ &\sim \sum_{g+1 \leq h \leq K} \frac{1}{h} f_H(h), \quad 0 \leq g \leq K-1 \end{aligned}$$

whereupon substitution of (5.2) yields (5.3). As noted above, the h gap sizes that can match up with a given HOL item size h are equally likely, so (5.4) follows directly from $\pi_{gh} \sim f_H(h)/h$ and (5.2). ■

5.4 Asymptotic Theory of Fragmentation

The general objective of this section is an estimate of the extent of fragmentation in statistical equilibrium. Specifically, in the state notation of Section 5.2, the goal is a large- M estimate of the equilibrium expected total number of fragments, $|L_1| + \dots + |L_N|$, for fixed $K \geq 2$. Recall that an item of size $j > 1$ is said to be *completely fragmented* when it is allocated a completely fragmented part of \mathcal{R} —no two slots of its allocation are adjacent.

	NF				BFD			
M	10	100	1000	10000	10	100	1000	10000
\bar{N}	6.56829	66.5382	666.552	6666.56	6.56283	66.5806	666.659	6666.32
\bar{N}_1	3.47090	33.4098	333.441	3330.45	3.46030	33.4949	333.651	3332.98
\bar{N}_{21}	0.79380	0.88536	0.91478	0.91675	1.56803	1.95471	2.05519	1.94872
\bar{N}_{22}	2.30359	32.2430	332.197	3332.19	1.53450	31.1310	330.953	3331.40

Table 5.1: Simulation results using the NF and BFD policies with $K = 2$, item size probabilities $q_1 = q_2 = 1/2$, and varying M : steady-state averages of the number of unfragmented size-2 items (N_{21}) compared to averages of the total number of items (N), the number of size-1 items (N_1), and the number of fragmented size-2 items (N_{22}).

In this section, we prove limit laws which show that for large M nearly all of the items receive completely-fragmented allocations. Before getting into the theory, we look very briefly at some experiments that brought out initial insights (experiments not oriented to asymptotic behavior have been reported in the past (e.g., in [38])). The analysis that follows begins with the case $K = 2$ and accommodates any item-size distribution on $\{1, 2\}$. The next subsection analyzes the case for general $K \geq 2$ and $q_1 > 0$, and the final subsection covers the general case $K \geq 2$ without the constraint that q_1 be positive.

The analysis in this section applies to the fragmentation process in statistical equilibrium: the focus is on large- M asymptotics with $\{q_1, \dots, q_K\}$ fixed. To improve readability, we will occasionally omit “ $M \rightarrow \infty$ ” when the limit is clear in context.

5.4.1 Preliminaries

For the experimental results in this subsection, the simulation of 10^8 departures produced in all cases a behavior that was indistinguishable from stationarity. Additional details regarding the simulation tool are provided in 5.7.1.

Table 5.1 gives experimental results for $K = 2$ that describe behavior at departure epochs in the case where size-1 and size-2 items are equally probable. The values shown in the table for the quantities N_1 , N_{21} , and N_{22} are the respective experimental results for the numbers of size-1

	NF			BFD		
M	$\bar{\rho}_{2,M}$	$\bar{\rho}_{5,M}$	$\bar{\rho}_{10,M}$	$\bar{\rho}_{2,M}$	$\bar{\rho}_{5,M}$	$\bar{\rho}_{10,M}$
1000	0.99698	0.94927	0.68151	0.993957	0.825870	0.362162
2000	0.99846	0.97422	0.82478	0.997017	0.902945	0.522318
3000	0.99893	0.98248	0.87911	0.998075	0.933195	0.607607
4000	0.99917	0.98676	0.90714	0.998416	0.948368	0.663502
5000	0.99932	0.98927	0.92475	0.998898	0.958309	0.704494
6000	0.99940	0.99096	0.93668	0.999000	0.964694	0.734542
7000	0.99946	0.99188	0.94572	0.999180	0.969729	0.757376
8000	0.99949	0.99284	0.95187	0.999202	0.973263	0.776753
9000	0.99952	0.99340	0.95694	0.999324	0.976383	0.794857

Table 5.2: Simulation results using the NF and BFD policies with item sizes uniformly distributed and varying K , M : steady-state probability that an item of size- K is completely fragmented, $\rho_{K,M}$.

items, unsplit size-2 items, and size-2 items that split into 2 fragments placed into non-adjacent slots. It can be seen that, while the average numbers of items, \bar{N} , size-1 items, \bar{N}_1 , and split size-2 items, \bar{N}_{22} , grow linearly in M , no sustained growth at all is present in the average number \bar{N}_{21} of unsplit size-2 items. The approximate asymptotic value of 2 for \bar{N}_{21} under BFD is in fact exact; a more general result is proved below.

Detailed complete-fragmentation behavior was also studied in experiments for larger K with a focus on the uniform distribution on $\{1, \dots, K\}$. The results for a number of values of K are illustrated in Table 5.2 for NF and BFD, where $\bar{\rho}_{K,M}$ denotes the experimental estimate of the steady-state probability that size- K items are given a completely fragmented allocation of K non-contiguous slots. The trend toward 1 is monotone increasing toward complete fragmentation in all three cases, $K = 2, 5, 10$. Below, we prove an even stronger result than convergence in probability, but precise formulas have been out of reach so far. Note the comparison in the tables between NF and BFD whereby the items under the NF policy fragment significantly faster than under the BFD policy.

5.4.2 $K = 2$

A precise asymptotic formula is available for $K = 2$ and it requires only that the two item-size probabilities be positive. In deriving this formula, which measures the asymptotic expected number of unsplit 2's at departure times, our approach balances the equilibrium rate at which the number of unsplit 2's increases at departure times against the equilibrium rate at which it decreases at these times. Choose a departure at random in statistical equilibrium, and suppose it occurs at time t . We continue with the notation for N_{ij} from the previous subsection (also described in 5.7.2). Recall that the stable gap-wait states for $K = 2$ are $(0,1)$, $(0,2)$, and $(1,2)$, and note immediately that, in the gap-wait state transition at time t , $N_2(t)$ can change by at most 1.

Observation 1. *Suppose first that the departure occurs in a $(0,2)$ state. If the departure is a 1, then the state transitions to a $(1,2)$ -state and there is no change in $N_{21}(t)$. If the departure is a 2, then the HOL 2 will receive a split or unsplit allocation according as the departing 2 had a split or unsplit allocation, respectively. Again, there is no change in $N_{21}(t)$, so we may restrict the analysis to transitions out of states $(0,1)$ and $(1,2)$.*

Observation 2. *In a $(0,1)$ state, there can be no increase in $N_{21}(t)$ in any of the 3 possible gap-wait state transitions. There can be a decrease, but only by 1, and this happens if and only if $N_{21}(t) > 0$ and the departure is an unsplit 2. Then, since the rate parameter of the exponential residence times is 1, the rate of decrease in the number of unsplit 2's in transitions out of $(0,1)$ states is simply $N_{21}(t)$. It follows that the average rate at which $N_{21}(t)$ decreases from $(0,1)$ states is $\mathbb{E}'[N_{21}]$, where the prime indicates that the average is taken in equilibrium over all $(0,1)$ states.*

Observation 3. *No decrease in $N_{21}(t)$ need occur in $(1,2)$ states, since, as in the case of $(0,2)$ states, a departing unsplit 2 can (and will, if a largest-gap-first allocation policy is being used) be replaced by an unsplit 2. For an increase in $N_{21}(t)$ to occur, the next departure must be one of the at most two items using a slot adjacent to the empty slot. (Conceivably, there is a split 2 whose two slots are on either side of the gap, so that there is only one item whose departure can increment $N_{21}(t)$.) Thus $N_{21}(t)$ is incremented at rate at most 2.*

Since in the large- M limit the ratio of the equilibrium frequencies of (0,1)-states to (1,2)-states is $q_1/(1-q_1)$, equilibrium demands that $\mathbb{E}[N_{21}]q_1 \leq 2(1-q_1)$ giving $\mathbb{E}[N_{21}] \leq 2(1-q_1)/q_1$. Having observed that changes in N_{21} are rare, in the sense that they occur from any state with probability at most $2/M \rightarrow 0$, we conclude that the asymptotic equilibrium expected value $\mathbb{E}[N_{12}]$ of N_{12} over all states is equal to our $\mathbb{E}'[N_{21}]$. Hence, we have the following:

Theorem 4. *In the $K = 2$ case with positive item probabilities, the equilibrium expected value of the number of unsplit items of size 2 satisfies*

$$\mathbb{E}[N_{21}] \leq 2(1 - q_1)/q_1$$

asymptotically as $M \rightarrow \infty$, regardless of the allocation policy.

Note that, within the constraints of our allocation policies, not even a clairvoyant policy, i.e., one that can see the queue of arrivals and departures for the infinite future, can escape the extreme fragmentation guaranteed by Theorem 4.

If a policy scans the gaps in decreasing order of size, then an argument similar to that given in the proof of Theorem 4 can be used to bound by a constant the expected number of split 2's whose fragments are separated by only a single unoccupied slot (cf. Observation 3). From this there follows:

Corollary 1. *For any largest-gap-first policy in the $K = 2$ case, the asymptotic equilibrium rate at which $N_{21}(t)$ is incremented is precisely 2, and thus, the bound in Theorem 4 becomes*

$$\mathbb{E}[N_{21}] \sim 2(1 - q_1)/q_1, \quad M \rightarrow \infty.$$

Although the bound of Theorem 1 applies to the NF algorithm, Corollary 1 does not, since NF is not a largest-gap-first policy (there are transitions when an HOL 2 is not put into a gap of size 2, even when one exists). The problem of finding a tight bound for NF is much more difficult, even in the large- M asymptotic regime. The difficulty arises primarily in characterizing the behavior of transitions into and out of state (1,2). This can be expected because a new variable enters the analysis: the index of the position where the scan is to recommence relative to the positions of vacant slots.

5.4.3 $K \geq 2$ and $0 < q_1 < 1$

For this case, our argument is based on the notion of *bonds*, with a bond being any pair of adjacent slots that are either both empty or both occupied by the same item, i.e., a pair of adjacent slots within a gap or an item fragment. $B(t)$ denotes the total number of distinct bonds at time t . Note that in the $K = 2$ case, $B(t) = N_{21}(t)$.

We will show that, as long as there are some items of size 1, the expected number of bonds in equilibrium is bounded by a constant independent of M . To do this we will show, much as in the $K = 2$ case, that bonds are destroyed at a rate proportional to $B(t)$, but created at only a constant rate.

Observation 4. *Bonds can be created only when a departing item vacates one or more slots adjacent to an already present gap. In the worst case, a departure is about to take place while there are already $K-1$ single-slot gaps. In that case every pair of adjacent slots, exactly one of which is currently unoccupied, will become a bond at rate 1, so the expected increase in the number of bonds is $2K-2$. For example, if the gaps and a certain item of size K happen to be interleaved to create a giant interval of size $2K-1$, then that item's departure is the only one to create bonds, but it creates $2K-2$ of them. If, more typically, the gaps are spread out and every one of the $2K-2$ slots adjacent to a gap belongs to a different item, then only one new bond will be created but at rate $2K-2$. In summary, bonds are created at an average rate of at most $2K-2$.*

Theorem 5. *Suppose the probability of size-1 items satisfies $0 < q_1 < 1$. Then in equilibrium, the number of items only partially fragmented is bounded by a constant independent of the number M of slots, regardless of the allocation policy.*

Proof. In view of Observation 4, we need a suitable bound on the rate of bond destruction to complete the proof. Bonds are destroyed (but only one at a time, assuming a best-fit allocation policy) whenever a departing item has a fragment which is partly but not completely filled by an incoming item. Since no item can contain more than $K-1$ bonds, $B(t)$ is at most $K-1$ times the number of items that contain one or more fragments of size at least 2. Suppose that at a

moment when there are no gaps an item of size j departs, leaving at least one gap of size $i > 1$. If the next $j-i+1$ items are all ones, at least one of them will be placed in that gap, destroying a bond. In the worst case, $j = K$ and $i = 2$, but even then the rate of bond destruction is at least the product of 1) the probability of $K-1$ consecutive 1's, 2) the lower bound $B(t)/(K-1)$ on the number of items with at least one fragment of size 2 or greater, and 3) the probability of a full state (i.e., a state with no open slots). That is, we can write as a lower bound to the destruction rate:

$$q_1^{K-1} \times \frac{B(t)}{K-1} \times f_G(0)$$

where, from Eq. (5.3), $f_G(0) \geq 1/K$ is the limiting probability of a full state. Thus, the mean bond destruction rate is at least $CB(t)$ for some constant $C > 0$ independent of M .

We now finish the argument as before. Since at equilibrium the average rates of increase and decrease of $B(t)$ must be the same, we have that, from Observation 4 and the above argument, $2K-2 \geq C\mathbb{E}[B]$ and thus $\mathbb{E}[B] \leq (2K-2)/C$. ■

A note of warning may be in order here. In the proof, the constant provided by the theorem may be exponential in K with base $1/q_1$, suggesting that in practice M may need to be *much* larger than K to get close to complete fragmentation. This is indeed the case, but we defer illustrations along with further discussion to the next section on convergence issues.

Note that, if the item sizes with positive probability have a common divisor $d > 1$, then a best-fit allocation policy will treat the resource as if it were partitioned into unbreakable intervals of d slots each, with (perhaps) some slots left over that are never used. So no fragmentation beyond size d can be expected.

5.4.4 $K \geq 2$

The result in this section is completely general, but weaker than each of the preceding theorems. This is because of the case yet to be covered: the item sizes having positive probability do not include size 1 and are not all divisible by some $d > 1$. The analysis of this case follows and will be seen to be significantly more subtle.

It remains true that for large M , when an HOL item requesting j slots is waiting, states

with $0, 1, 2, \dots$, up to $j-1$ unoccupied slots are equally likely. Moreover, the argument of Observation 4 remains in effect: the average rate at which bonds are created cannot exceed $2K-2$. It is no longer generally the case, however, that bonds are destroyed at a rate proportional to $B(t)$. Thus, we cannot show $\mathbb{E}[B]$ is bounded by a constant. However, it is still possible, as shown below, to prove nearly complete fragmentation in the sense that $\lim_{M \rightarrow \infty} \mathbb{E}[B]/M = 0$.

The difficulty is that even when a gap of size 2 or more is present, it may be that no sequence of items can force the destruction of a bond. For example, suppose item sizes are 2 and 3, all items are currently split, but items of size 3 are split into only two pieces. Then a departing 3 will cause a gap of size 2 to appear, but a best-fit allocation algorithm will fill it immediately using the HOL item, so no bond is destroyed.

Indeed, it is not obvious that bonds will ever get destroyed in this situation; can it be that no further fragmentation will occur? Not quite—the reason is that this situation, with all bonds inside items of size 3, is unstable. Bonds will migrate to items of size 2 and eventually two whole items of size 2 will depart with an HOL 3 waiting, causing the destruction of a bond.

To argue that such an event happens in general, we say that a currently active item is of *type* $\langle i, j \rangle$ if it is for i slots and contains exactly j bonds. A type $\langle i, j \rangle$ is said to be *rife* if the expected equilibrium number of items of type $\langle i, j \rangle$ is bounded below by CM , for some constant factor $C > 0$ and infinitely many values of M . The idea will be to show, by contradiction, that no type $\langle i, j \rangle$ with $j > 0$ can be rife.

To view the dynamic resource allocation process as a continuous-time Markov chain, we imagine that there is a cupboard of M clocks, which “ring” independently and periodically after exponential mean-1 waiting times. When an item is allocated it takes a clock from the cupboard and when the clock rings, the item departs, putting the clock back in the cupboard. When a clock in the cupboard rings, nothing happens. We will make use of the fact that if $\langle i, j \rangle$ is rife, with constant factor C , then for any fixed n and infinitely many M the probability σ that, at a given time t in equilibrium, the next n rings will all signal departures of items of type $\langle i, j \rangle$ is bounded below by $\frac{1}{2}C^n$. To see that this is so, let p_k , $k = 0, 1, 2, \dots$, denote the probability

that, in equilibrium, exactly k items of type $\langle i, j \rangle$ are active. Then,

$$\sigma \geq \sum_{k=0}^M p_k \prod_{h=0}^{n-1} (k-h)/M \sim \sum_{k=0}^M p_k (k/M)^n$$

and so the result follows from

$$\sum_{k=0}^M p_k (k/M)^n \geq \left(\sum_{k=0}^M p_k \frac{k}{M} \right)^n \geq C^n.$$

Observation 5. Suppose that some state $\langle i, j \rangle$ with $j > 0$ is rife; we assert that there is another rife state $\langle i', j' \rangle$ with $i'/j' < i/j$. (Of course, since there are only finitely many pairs $\langle i, j \rangle$ with $0 < i \leq K$ and $0 < j < K$, this state of affairs is not possible.) To prove the assertion, note that the fraction i/j exceeds 1 and thus cannot be a common divisor of every item size. Let i' be an item size (with $q_{i'} > 0$) such that $i'/(i/j) = i'j/i$ is not an integer. Suppose that at time t all slots are occupied while i items of size i' are waiting; and at the next i' rings, all signal departures are of items of type $\langle i, j \rangle$. The probability of such an event is at least $q_{j'}^i C^{i'}/(2KM)$, for infinitely many M , assuming C is as in the definition of “rife” for type $\langle i, j \rangle$. If half or more of the times this happens some bond is destroyed, the bond destruction rate would exceed the constant $2K-2$ bond creation rate, a contradiction. It follows that most of the time all $i'j$ bonds in the departing items are distributed among the i incoming items of size i' . Suppose that on this occasion b_k bonds end up in the k th item. If $i'/b_k > i/j$ for each k , $1 \leq k \leq i$, then

$$i'j = \sum_{k=1}^i b_k = i' \sum_{k=1}^i b_k/i' < i'i(j/i) = i'j,$$

a contradiction. But i'/b_k cannot be equal to i/j because then $i' = b_k i/j$, an integer multiple of i/j . We conclude that for some k , $i'/b_k < i/j$. Since there are fewer than K possible values for this b_k , items of some type $\langle i', j' \rangle$ with $i'/j' < i/j$ are being created with probability at least

$$q_{j'}^i C^{i'}/(4K^3)$$

and therefore at rate M times this constant. So this new type is rife, proving the assertion.

We have shown that there are no rife types $\langle i, j \rangle$ with $j > 0$. If the equilibrium expected fraction of bonds $\mathbb{E}[B]/M$ were not bounded by some constant C , some type $\langle i, j \rangle$ with $j > 0$ would have to have equilibrium expected frequency at least CM/K^2 and thus be rife. We have proved the following by contradiction.

Theorem 6. *Suppose the item size probabilities are such that the values j for which $q_j > 0$ have no nontrivial common divisor. Then in equilibrium, the fraction of items partially fragmented tends to 0 as $M \rightarrow \infty$, regardless of allocation policy.*

The caveat following Theorem 5 applies even more strongly in this case. Bounds are not likely to be useful in a real-life problem—the time needed for nearly complete fragmentation is going to be even more astronomical than before.

5.5 Convergence

So far, we have studied the extent of fragmentation in statistical equilibrium for large M/K . We have yet to study the transient behavior of the fragmentation process and to get a more quantitative appreciation of the properties of this process as a function of the parameters. Unsurprisingly, an analysis of convergence rates is difficult, so we are confined primarily to experiments and suggestive examples in what follows. Our space constraints prevent us from full coverage of our extensive experiments, so we limit ourselves to properties closely related to the theory of the preceding section.

We begin with a return to the red flag raised by the potentially huge constant derived in the proof of Theorem 5. We give an example in which the parameters are far from extreme, yet it illustrates the extremely slow convergence to statistical equilibrium that can occur in practice. As the example makes clear, the problem is that M has to be taken enormously large (far beyond realizable values) to ensure convergence to states where nearly all items are completely fragmented. We follow this example in the second subsection with a more general assessment of convergence to statistical equilibrium, beginning as earlier with the simplest non-trivial case $K = 2$. We then illustrate our study of cases $K \geq 2$ by specializing to uniform item-size

distributions on $\{1, \dots, K\}$. In the third subsection, we direct our attention to the constant bounds on partially fragmented items guaranteed by Theorem 5. These bounds increase with K , and, considering Example 1 below, it is of immediate interest to get a good idea of the rate of increase.

5.5.1 An Instructive Example

Consider the implication of the proof of Theorem 5 that there exists a constant C such that, for all M sufficiently large, the number of partially fragmented items is at most C ; more specifically, consider estimates of “sufficiently large.” The following example shows that they can be egregiously large.

Example 1. Suppose that items are of size 1 and 9 with probabilities .1 and .9, respectively. Suppose further that we are close to complete fragmentation in that it has developed to a point where 90% of the items of size 9 have been completely fragmented, the rest having 7 fragments of size one and one fragment of size 2. Then there will typically be about 4 singleton gaps in play, and bonds will be created (assuming a best-fit allocation policy) at rate about 7. To destroy a bond, however, we need at least 8 size-1 items in a row following the departure of a partially-fragmented 9; this happens at rate about 10^{-8} times the number of such 9s, so if we want the fragmentation to continue forward, we need $10^{-8} \times (M/9)/10$ to be greater than 7, forcing $M > 10^{10}$. Change K to 99 or q_1 to .01, and M will need to exceed the estimated number of elementary particles in the universe. (Note that even the time to reach the initial condition of the example is very likely to be huge.)

Fig. 5.2 shows the simulation results that estimate the steady-state fraction of size-9 items that are completely fragmented under the BFD algorithm as a function of M for several choices of q_1 . The values of M required for nearly complete fragmentation increase by roughly an order of magnitude as q_1 decreases from one value to the next in the sequence 0.9, 0.7, 0.5, 0.3, 0.1. However, consistent with the example above, the required values of M for nearly complete fragmentation grow much faster as q_1 decreases below 0.1. Convergence to complete fragmentation for smaller q_1 appears to require exponentially larger M . In particular, the curve for $q_1 = 0.1$ is still substantially short of complete fragmentation and was found to remain so until $M \approx 10^{10}$.

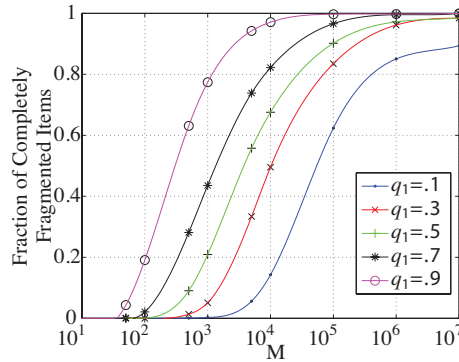


Figure 5.2: Steady-state fraction of completely fragmented items as a function of M under the BFD algorithm, $K = 9$, and a parameter $q_1 = 1 - q_9$.

Similar results will hold for any *best-fit* algorithm.

5.5.2 Convergence to Nearly Complete Fragmentation

We begin as in the previous section with $K = 2$, the simplest non-trivial case. We then illustrate our experimental results for general K with an example describing the convergence to complete fragmentation with item size a parameter.

5.5.2.1 $K = 2$

For this case, Theorem 4 and Corollary 1 give explicit asymptotic results for the constant bound on the number of unsplit 2's in statistical equilibrium. Might we hope to be equally successful in an analysis of convergence rates from an unfragmented initial state? For the moment, no, but a rough calculation is available which leads to an interesting formula that is very accurate when q_1 and hence q_2 are not too close to 0 or 1, and M is large.

We adopt the BFD algorithm for this case, and consider the number N_{22} of split 2's in the chain embedded at the epochs where a 2 departs in gap-wait state (0,1). These are the only epochs of the fragmentation process where N_{22} can increase; an HOL unsplit 2 becomes a split 2 with a probability proportional to the number of unsplit 2's. We ignore increases in the number of unsplit 2's since these events, only in gap-wait state (1,2), occur with probability $O(1/M)$. The quantity to be estimated is $E[T_\gamma]$, $\gamma \in (0, 1)$, which tells how long it takes on average for the

	$q_1 = 0.1$			$q_1 = 0.9$		
M	$\mathbb{E}[T_\gamma]$	\bar{T}_γ	Error	$\mathbb{E}[T_\gamma]$	\bar{T}_γ	Error
1000	1090.70	1221.20	10.69%	209.33	221.40	5.45%
10000	10906.98	11115.90	1.88%	2093.26	2054.30	1.90%
100000	109069.82	109091.00	0.02%	20932.59	20942.20	0.05%

Table 5.3: Accuracy of approximate convergence times for $K = 2$.

embedded fragmentation process to increase the number of split 2's to a fraction γ of the original number of 2's, all of which were unsplit. Note that the embedded process is an occupancy type process. The calculations estimating $\mathbb{E}[T_\gamma]$, $\gamma \in (0, 1)$, are routine and can be found in 5.7.3. The result is,

$$\mathbb{E}[T_\gamma] \approx M \frac{1 - q_1}{2 - q_1} \ln \frac{1}{1 - \gamma}.$$

The accuracy of this formula is illustrated in Table 5.3.

Out of a total of D events, the average number of events in the embedded process is asymptotically the total number of events times the probability $\pi_{01} = q_1/(2 - q_1)$ of the (0,1) state (cf. Theorem 3) times the probability $1 - q_1$ of a 2 departure. Then $\mathbb{E}[T_\gamma]/[(1 - q_1)\pi_{01}]$ approximates the asymptotic expected total number $\mathbb{E}[D_\gamma]$ of departures needed to fragment a fraction γ of the unsplit 2's of the initial state. This yields

$$\mathbb{E}[D_\gamma] \approx \frac{M}{q_1} \ln \frac{1}{1 - \gamma},$$

which has an accuracy comparable to that for $\mathbb{E}[T_\gamma]$.

5.5.2.2 General $K \geq 2$

Example 1 above suggests that questions with realistic answers concern the time to approach states that are just moderately close to complete fragmentation. To get a better handle on the term "moderately close," we resorted to experiments, in which the item-size distribution of choice was the uniform law on $\{1, \dots, K\}$. The results of a typical experiment are shown in Fig. 5.3 where the curves represent the growth in average fragmentation normalized by the

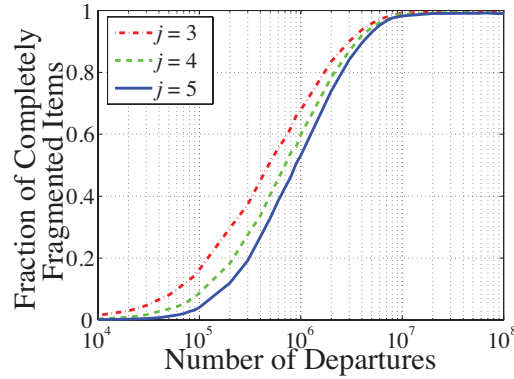


Figure 5.3: Transient of the normalized average number of fragments per item for size- j requests under BFD (log scale): $K = 5$ (uniform distribution), $M = 100,000$.

number of fragments for complete fragmentation, i.e., the item size. The region of near linear growth in the scaling of Fig. 5.3 shows that even the peak rate of fragmentation is quite slow, in particular approximately logarithmic in the number of departures. Intuitively, the larger item sizes can be expected to approach high levels of fragmentation more slowly the smaller size items. But until the fragmentation of the larger items becomes well advanced, the smaller items may well be inhibited from fragmenting by the relatively large gaps made available by departing, coarsely fragmented large items. Accordingly, the difference in convergence rates across the item sizes is not large.

5.5.3 Estimates of the Number of Partially Fragmented Items from Theorem 5

Recall that, for $q_1 > 0$, Theorem 5 guarantees that, in statistical equilibrium, there is at most a constant number—independent of M —of items which are partially fragmented (i.e., containing bonds). To understand the scaling of this constant as a function of the item size distribution, we resorted to numerical experiments. Fig. 5.4 presents the steady-state number of bonds using BFD when item sizes are uniformly distributed on $\{1, \dots, K\}$. A ratio M/K of resource size to maximum item size was selected so as to ensure that the numbers of bonds had leveled off for all K ; there was no discernible trend when the simulations terminated.

The experiments were quite demanding in the time required as K was increased. The figure shows results only for the constants corresponding to $K \leq 10$. The data were fit to a cubic polynomial. Echoing Example 1, if K were increased by any significant amount, the constant bound was effectively unreachable with any value of M that would pass a reality check.

5.6 Conclusions and Future Work

File allocation in disk-based storage and dynamic spectrum access are but two applications in which requests for a resource are allowed to be fragmented in the interest of more efficient utilization. The main contribution of this work is the analysis of the fragmentation process in a resource operating at capacity. Despite over four decades of research in the area of dynamic storage allocation, this is the first attempt to study this problem.

Theorems 4, 5, and 6 constitute our contributions to the theory of fragmentation and show that for general item size distributions with large values of the resource size, nearly all items become *completely fragmented* in statistical equilibrium. In addition, we find a precise asymptotic formula for the number of partially fragmented items in the baseline case where requests are limited to size-1 or size-2. However, precise asymptotic formulas for the number of non-completely fragmented items have been out of our reach so far for arbitrary values of K . We supplement the analysis through examples and experimental results which show that the rates of convergence can be surprisingly slow; even the time required to become *moderately close* to states with nearly all items completely fragmented can be extremely long.

This leaves a number of interesting open problems, among them being more extensive studies of algorithms and of fragment configurations. Needless to say, the methodology of such studies will likely be limited to experimental approaches. An alternative source of open problems arises from generalizations of our assumptions, all of which are unlikely candidates for an analysis leading to explicit formulas. For example, the exponential assumption gives critical simplifications to analysis, but changes in behavior resulting from other residence-time distributions are worth investigating. In addition, instead of a system operating at capacity, in which there is always an item waiting, one could adopt an underlying, fully stochastic model of demand, e.g., a Poisson

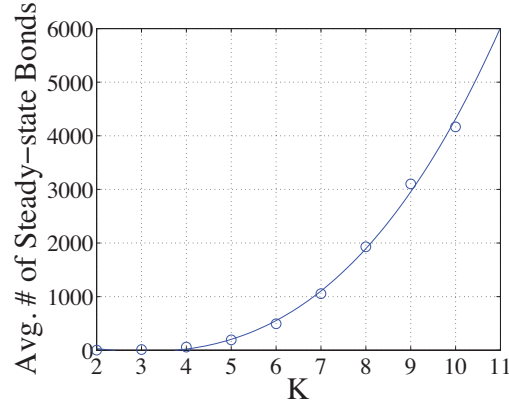


Figure 5.4: Average number of bonds in equilibrium under BFD, with $M/K = 10^5$, item sizes uniformly distributed on $\{1, \dots, K\}$.

arrival process. Finally, more realistic models would relax the independence assumptions. A prime example appropriate for Dynamic Spectrum Access applications would allow residence times to depend on fragmentation. One may have to assume that the greater the fragmentation of an item, the longer its residence time.

5.7 Appendices

5.7.1 Simulation Tool

The numerical experiments reported in this chapter were conducted using a discrete-event simulator developed in C++. The simulations were executed until the system was reasonably close to steady-state at which point we randomly sampled for the performance measure of interest. The excellent accuracy of the tool was established in tests against the continuous version of this model studied in [38]. We also tested the accuracy of the tool against results in [92] where the model did not allow fragmentation (cf. Theorem 1). We found that our simulation results were within 1% of the analytical results.

5.7.2 Notation

\mathcal{R}	Resource
M	Size of \mathcal{R} (in slots)
K	Maximum request size (in slots)
U	Random variable denoting the request size
q_j	Equivalent to $f_U(j)$, the request size pmf
u_i	i^{th} realization of U
$H(t)$	The HOL request size at time t
$G(t)$	The total number of available slots at time t
N	Random variable denoting the number of requests being served by \mathcal{R}
N_i	Random variable denoting the number of requests of size i
N_{ij}	Random variable denoting the number of requests of size i split into j fragments
N_g	Random variable denoting the number of distinct gaps in \mathcal{R}
$\rho_{K,M}$	Steady-state probability of total-fragmentation for resource size M , max request size K
$B(t)$	Total number of distinct bonds at time t
α	The parameter of the continuous model. $K, M \rightarrow \infty$ subject to $K/M \rightarrow \alpha$, with $\alpha \in (0, 1]$
T_γ	The number of departure epochs required to increase N_{22} to γN_2

5.7.3 Rough Estimate of $\mathbb{E}[T_\gamma]$

Recall that the epochs of the embedded fragmentation process occur in the $(0,1)$ gap-wait state when a 2 departs. These are the epochs where the number of split 2's can increase. T_γ is the number of such epochs needed to change a fraction γ of the unsplit 2's to split 2's, assuming that all 2's are unsplit in the initial state. To carry the analysis forward, we reduce this occupancy-type process to its elementary form by taking N_2 to be constant at its mean $\beta \equiv E[N_2]$, which will give a useful asymptotic approximation in that region of the parameter space where the variation of N_2 around its mean is relatively small. Within the simpler model, the time to split a fraction γ of the original β unsplit 2's will be the sum of $\lfloor \gamma\beta \rfloor$ independent geometric random variables T_i with parameters $(\beta - i + 1)/\beta$, $i = 1, 2, \dots, \lfloor \gamma\beta \rfloor$, and mean values $\beta/(\beta - i + 1)$, where T_i is the number of random draws from the set of 2's that have to be made before getting one that's not yet split, given that there are exactly $i - 1$ that have already been split. For

simplicity, let $\gamma\beta$ be an integer in what follows. We have, when the number β of size-2 items is large,

$$\begin{aligned} \sum_{1 \leq i \leq \gamma\beta} E[T_i] &= \sum_{1 \leq i \leq \gamma\beta} \frac{\beta}{\beta - i + 1} \\ &= \beta \sum_{k=\beta(1-\gamma)+1}^{\beta} \frac{1}{k} \approx \beta \ln \frac{1}{1-\gamma} \end{aligned}$$

A routine calculation shows that $\beta \sim M(1 - q_1)/(2 - q_1)$, and so finally

$$E[T_\gamma] \approx M \frac{1 - q_1}{2 - q_1} \ln \frac{1}{1 - \gamma}$$

This page intentionally left blank.

Chapter 6

Conclusions

In this thesis, we considered wireless resource allocation problems for the Internet of Things (IoT).

In Chapters 2 and 3, we studied the Internet of Tags (IoTags). First we presented a prototype IoTag, termed an Energy Harvesting Active Networked Tag (EnHANT). By developing a first-of-its-kind testbed, we evaluated protocols on all layers of the stack for networks of EnHANTs. Then, we took a top-down approach to develop Panda, an ND protocol for energy harvesting nodes. Using techniques from renewal theory, we analyzed and optimized the discovery rate of Panda subject to a power budget. Through experiments using a COTS energy harvesting node, we demonstrated the practicality of the protocol as well as the improved (up to 3x) performance over related works.

In Chapter 4, we developed the Predictive Finite-horizon PF Scheduling ((PF)²S) Framework that exploits predictable mobility in cellular networks. We presented *extensive channel measurement* results from a 3G network and characterized mobility-induced channel state trends. Our *trace-based simulations* consider of the (PF)²S Framework indicate that the framework can increase the throughput by 15%–55% compared to traditional PF schedulers, while improving fairness.

In Chapter 5, we studied fragmentation within a probability model of the combinatorial structures. Despite over four decades of research in the area of dynamic storage allocation, this

is the first attempt to study this problem. We presented a suite of theorems to contribute to the theory of fragmentation and show that for general item size distributions with large values of the resource size, nearly all items become *completely fragmented* in statistical equilibrium. In addition, we showed that the rates of convergence can be surprisingly slow; even the time required to become *moderately close* to states with nearly all items completely fragmented can be extremely long.

In Appendix 6, we presented our contributions to *engineering education*. Over the last 5 years, we have engaged over 180 semester-long student projects within the EnHANTs project. Our experience demonstrated the feasibility of engaging diverse groups of students on large-scale interdisciplinary research efforts. It also shed light on some potential pitfalls of such efforts (e.g., inadequate cross-group communication and knowledge transfer), and suggests best practices to overcome these challenges.

In summary, this thesis contributed to the understanding of wireless resource allocation problems arising in the IoT. We evaluated resource allocation problems on all layers of the stack for IoTags. We presented a cellular scheduling framework to support intelligent transportation systems. Finally, we provided a fundamental analysis of fragmentation for dynamic spectrum access networks, with numerous IoT applications.

Bibliography

- [1] CVX: Matlab Software for Disciplined Convex Programming, v2.0 beta.
- [2] “EnOcean Energy Harvesting Wireless Sensor Solutions and Networks,” www.enocean.com/.
- [3] “MicroMote (M³) project,” <http://www.eecs.umich.edu/~prabal/>.
- [4] “Crossbow MICA2 Mote,” www.xbow.com/Products/Productpdfs/Wirelesspdf/MICA2Datasheet.pdf, 2009.
- [5] “Texas Instruments MSP430 Solar Energy Harvesting Development Tool,” focus.ti.com/docs/toolsw/folders/print/ez430-rf2500-seh.html, 2012.
- [6] “Energy harvesting active networked tags (EnHANTs) project,” <http://enhants.ee.columbia.edu>, 2014.
- [7] “Harlem children society,” harlemchildrensociety.org, 2014.
- [8] “Low-cost low-power 2.4 GHz RF transceiver (SWRS 040C),” *Texas Instruments, Inc.*, 2015.
- [9] “Texas Instruments CC2640 SimpleLink bluetooth smart wireless MCU,” ti.com/lit/ds/symalink/cc2640.pdf, 2015.
- [10] 3rd Generation Partnership Project, “3GPP specification detail: Physical layer procedures (FDD),” <http://www.3gpp.org/ftp/Specs/html-info/25214.htm>.

-
- [11] ABET, “Engineering Change: A study of the impact of EC2000,” www.abet.org/engineering-change, 2006.
 - [12] Y. Afsar, J. Sarik, M. Gorlatova, G. Zussman, and I. Kymissis, “Evaluating photovoltaic performance indoors,” in *Proc. 38th IEEE Photovoltaic Specialist Conference (PVSC38)*, June 2012.
 - [13] V. Aggarwal, H. Jana, R. Karloff, , R. Margolies, N. Shankaranarayanan, A. Sridharan, V. Vaishampayan, and G. Zussman, “System and method for performing cellular scheduling with channel quality predictions,” Sept. 2013, US Patent Pending.
 - [14] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, “CRAHNs: Cognitive radio ad hoc networks,” *Ad Hoc Networks*, vol. 7, no. 5, pp. 810–836, 2009.
 - [15] S. H. Ali, V. Krishnamurthy, and V. C. Leung, “Optimal and approximate mobility-assisted opportunistic scheduling in cellular networks,” *IEEE Trans. Mobile Comput.*, vol. 6, no. 6, pp. 633–648, June 2007.
 - [16] J. Andersen, T. Rappaport, and S. Yoshida, “Propagation measurements and models for wireless communications channels,” *IEEE Commun. Mag.*, vol. 33, no. 1, pp. 42–49, 1995.
 - [17] M. Andrews and L. Zhang, “Scheduling over nonstationary wireless channels with finite rate sets,” *IEEE/ACM Trans. Networking*, vol. 14, no. 5, pp. 1067–1077, Oct. 2006.
 - [18] M. Andrews, “A survey of scheduling theory in wireless data networks,” in *Wireless Communications*, ser. The IMA Volumes in Mathematics and its Applications. Springer, 2007, vol. 143, pp. 1–17.
 - [19] “Atmel AVR 8- and 32-bit Microcontrollers,” www.atmel.com/products/microcontrollers/avr/default.aspx, 2014.
 - [20] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey,” *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

-
- [21] Audio-technica, “Spectrapulse ultra wideband (UWB) wireless microphone system,” http://www.audio-technica.com/cms/wls_systems/e7dd603ca6147a97/index.html, 2012.
 - [22] N. J. August, H.-J. Lee, and D. S. Ha, “Enabling distributed medium access control for impulse-based ultrawideband radios,” *IEEE Trans. Veh. Technol.*, vol. 56, no. 3, pp. 1064–1075, May 2007.
 - [23] N. August, H.-J. Lee, and D. Ha, “Pulse sense: a method to detect a busy medium in pulse-based ultra wideband (UWB) networks,” in *Ultra Wideband Systems*, May 2004.
 - [24] M. Bakht, M. Trower, and R. H. Kravets, “Searchlight: Won’t you be my neighbor?” in *Proc. ACM MobiCom’12*, Aug. 2012.
 - [25] S. Bandyopadhyay and A. P. Chandrakasan, “Platform architecture for solar, thermal, and vibration energy combining with MPPT and single inductor,” *to appear, IEEE J. Solid-State Circuits*, 2012.
 - [26] H. Bang, T. Ekman, and D. Gesbert, “Channel predictive proportional fair scheduling,” *IEEE Trans. Wireless Commun.*, vol. 7, no. 2, pp. 482–487, Feb. 2008.
 - [27] P. Barford, J. Kline, D. Plonka, and A. Ron, “A signal analysis of network traffic anomalies,” in *Proc. ACM SIGCOMM’02*, 2002.
 - [28] A. Bernat, P. J. Teller, A. Gates, and N. Delgado, “Structuring the student research experience,” in *Proc. ACM ITiCSE’00*, July 2000.
 - [29] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1999.
 - [30] M. E. Bhalke, S. P. Subbarao, and I. Kymissis, “A laboratory process for encapsulation of air sensitive organic devices,” *IEEE Trans. Electron Devices*, vol. 57, no. 1, pp. 153–156, Jan. 2010.
 - [31] S. Borst, N. Hegde, and A. Proutiere, “Mobility-driven scheduling in wireless networks,” in *Proc. IEEE INFOCOM’09*, Apr. 2009.

-
- [32] T. Bu, L. Li, and R. Ramjee, “Generalized proportional fair scheduling in third generation wireless data networks,” in *Proc. INFOCOM’06*, 2006.
 - [33] G. Chandrasekaran, T. Vu, A. Varshavsky, M. Gruteser, R. Martin, J. Yang, and Y. Chen, “Tracking vehicular speed variations by warping mobile phone signal strengths,” in *Proc. IEEE PerCom’11*, Mar. 2011.
 - [34] S. Chen, P. Sinha, N. Shroff, and C. Joo, “Finite-horizon energy allocation and routing scheme in rechargeable sensor networks,” in *Proc. IEEE INFOCOM’11*, Apr. 2011.
 - [35] Z. Chen, T. See, and X. Qing, “Small printed ultrawideband antenna with reduced ground plane effect,” *IEEE Trans. Antennas Propag.*, vol. 55, no. 2, pp. 383–388, 2007.
 - [36] E. Cinlar, “Markov renewal theory,” *Advances in Applied Probability*, vol. 1, no. 2, pp. 123–187, 1969.
 - [37] E. Coffman, R. Margolies, P. Winkler, and G. Zussman, “Fragmented random structures,” *presented at Analysis of Algorithms 2014 (AofA’14) Poster Session*, June 2014.
 - [38] E. Coffman, P. Robert, F. Simatos, S. Tarumi, and G. Zussman, “A performance analysis of channel fragmentation in dynamic spectrum access systems,” *Queueing Systems, special issue of selected papers from ACM SIGMETRICS’10*, vol. 71, no. 3, pp. 293–320, July 2012.
 - [39] E. Coffman and T. Leighton, “A provably efficient algorithm for dynamic storage allocation,” *J. Comp. Sys. Sci.*, vol. 38, no. 1, pp. 2 – 35, 1989.
 - [40] E. Coffman, R. Margolies, P. Winkler, and G. Zussman, “Performance evaluation of fragmented structures: A theoretical study,” *Performance Evaluation*, vol. 79, pp. 273 – 286, Sept. 2014, Special Issue from IFIP Performance 2014.
 - [41] B. Coleman and M. Lang, “Collaboration across the curriculum: A disciplined approach to developing team skills,” in *Proc. ACM SIGCSE’12*, Feb. 2012.

-
- [42] M. Cong, K. Kim, M. Gorlatova, J. Sarik, J. Kymissis, and G. Zussman, “CRAWDAD data set columbia/kinetic (v. 2014-05-13),” <http://crawdada.org/columbia/kinetic/>, May 2014.
 - [43] “Contiki,” <http://www.contiki-os.org/>, 2014.
 - [44] M. Crepaldi, C. Li, K. Dronson, J. Fernandes, and P. Kinget, “An interference robust S-OOK IR-UWB transceiver chipset,” *IEEE J. Solid-State Circuits*, vol. 46, no. 10, pp. 2284–2299, Oct. 2011.
 - [45] F. Cuomo, C. Martello, A. Baiocchi, and F. Capriotti, “Radio resource sharing for ad hoc networking with UWB,” *IEEE J. Sel. Areas Commun.*, vol. 20, no. 9, pp. 1722–1732, Dec. 2002.
 - [46] D. Daly, P. Mercier, M. Bhardwaj, A. Stone, Z. Aldworth, T. Daniel, J. Voldman, J. Hildebrand, and A. Chandrakasan, “A pulsed UWB receiver SoC for insect motion control,” *IEEE J. Solid-State Circuits*, vol. 45, no. 1, pp. 153–166, Jan. 2010.
 - [47] N. Dang, E. Bozorgzadeh, and N. Venkatasubramanian, “Quares: Quality-aware data collection in energy harvesting sensor networks,” in *IEEE IGCC’11*, June 2011.
 - [48] A. Dementyev, J. Gummeson, D. Thrasher, A. Parks, D. Ganesan, J. Smith, and A. Sample, “Wirelessly powered bistable display tags,” in *Proc. ACM UbiComp’13*, 2013.
 - [49] M. Di Benedetto, L. De Nardis, G. Giancola, and D. Domenicali, “The aloha access (UWB)² protocol revisited for IEEE 802.15.4a,” *ST Journal of Research*, Mar. 2006.
 - [50] A. Dunkels, “The ContikiMAC radio duty cycling protocol,” Dec. 2011, Swedish Institute of Computer Science.
 - [51] P. Dutta, S. Dawson-Haggerty, Y. Chen, C.-J. Liang, and A. Terzis, “Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless,” in *Proc. ACM SenSys ’10*, Nov. 2010.

-
- [52] P. Dutta, J. Hui, J. Jeong, S. Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse, and D. Culler, "Trio: enabling sustainable and scalable outdoor wireless sensor network deployments," ser. IEEE IPSN '06, 2006.
- [53] P. Dutta and D. Culler, "Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications," in *Proc. ACM SenSys'08*, 2008.
- [54] C. L. Dym, A. M. Agogino, O. Eris, D. D. Frey, and L. J. Leifer, "Engineering design thinking, teaching, and learning," *Journal of Engineering Education*, vol. 94, no. 1, pp. 103–120, 2005.
- [55] A. El-Hoiydi and J.-D. Decotignie, "WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks," in *Proc. IEEE ISCC'04*, July 2004.
- [56] D. Evans, "The internet of things: How the next evolution of the internet is changing everything," *CISCO white paper*, vol. 1, 2011.
- [57] K.-W. Fan, Z. Zheng, and P. Sinha, "Steady and fair rate allocation for rechargeable sensors in perpetual sensor networks," in *Proc. ACM SenSys'08*, Nov. 2008.
- [58] R. M. Felder, D. R. Woods, J. E. Stice, and A. Rugarcia, "The future of engineering education ii. teaching methods that work," *Chemical Engineering Education*, vol. 34, no. 1, pp. 26–39, 2000.
- [59] M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. New York, NY, USA: W. H. Freeman & Co., 1990.
- [60] M. Gorlatova, P. Kinget, I. Kymissis, D. Rubenstein, X. Wang, and G. Zussman, "Energy-harvesting active networked tags (EnHANTs) for ubiquitous object networking," *IEEE Wireless Commun.*, vol. 17, no. 6, pp. 18–25, Dec. 2010.
- [61] M. Gorlatova, R. Margolies, J. Sarik, G. Stanje, J. Zhu, B. Vignraham, M. Szczodrak, L. Carloni, P. Kinget, I. Kymissis, and G. Zussman, "Prototyping energy harvesting active networked tags (EnHANTs)," in *Proc. IEEE INFOCOM'13 Miniconference*, Apr. 2013.

-
- [62] M. Gorlatova, Z. Noorbhaiwala, A. Skolnik, J. Sarik, M. Szczodrak, J. Chen, M. Zapas, L. Carloni, P. Kinget, I. Kymissis, D. Rubenstein, and G. Zussman, “Demo: Prototyping energy harvesting active networked tags,” *Demo presented in ACM MobiCom’10*, Sept. 2010.
 - [63] M. Gorlatova, J. Sarik, G. Grebla, M. Cong, I. Kymissis, and G. Zussman, “Movers and shakers: Kinetic energy harvesting for the internet of things,” in *Proc. ACM SIGMETRICS’14*, June 2014.
 - [64] —, “Movers and Shakers: Kinetic Energy Harvesting for the Internet of Things,” *IEEE J. Sel. Areas Commun., Special Issue on Wireless Communications Powered by Energy Harvesting and Wireless Energy Transfer*, vol. 33, no. 9, 2015.
 - [65] M. Gorlatova, J. Sarik, P. Kinget, I. Kymissis, and G. Zussman, “Project-based learning within a large-scale interdisciplinary research effort,” in *Proc. ACM ITiCSE’13*, July 2013.
 - [66] M. Gorlatova, T. Sharma, D. Shrestha, E. Xu, J. Chen, A. Skolnik, D. Piao, P. Kinget, I. Kymissis, D. Rubenstein, and G. Zussman, “Demo: Prototyping Energy Harvesting Active Networked Tags (EnHANTs) with MICA2 Motes,” in *Proc. IEEE SECON’10*, June 2010.
 - [67] M. Gorlatova, A. Wallwater, and G. Zussman, “Networking low-power energy harvesting devices: Measurements and algorithms,” in *Proc. IEEE INFOCOM’11*, Apr. 2011.
 - [68] —, “Networking low-power energy harvesting devices: Measurements and algorithms,” *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1853–1865, 2013.
 - [69] —, “Networking low-power energy harvesting devices: Measurements and algorithms,” *IEEE Trans. on Mobile Computing*, vol. 12, no. 9, pp. 1853–1865, Sept. 2013.
 - [70] —, “Networking low-power energy harvesting devices: Measurements and algorithms,” *IEEE Trans. Mobile Comput.*, vol. 12, no. 9, pp. 1853–1865, Sept. 2013.

-
- [71] M. Gorlatova, M. Zapas, E. Xu, M. Bahlke, I. J. Kymissis, and G. Zussman, "CRAWDAD data set columbia/enhants (v. 2011-04-07)," <http://crawdad.cs.dartmouth.edu/columbia/enhants>, Apr. 2011.
 - [72] T. Greening and J. Kay, "Undergraduate research experience in computer science education," in *Proc. ACM ITiCSE'02*, June 2002.
 - [73] J. Gummesson, S. S. Clark, K. Fu, and D. Ganesan, "On the limits of effective micro-energy harvesting on mobile CRFID sensors," in *Proc. ACM MobiSys'10*, June 2010.
 - [74] H. A. Hadim and S. K. Esche, "Enhancing the engineering curriculum through project-based learning," in *Proc. IEEE Frontiers in Education*, 2002.
 - [75] J. Hajipour and V. C. M. Leung, "Proportional fair scheduling in multi-carrier networks using channel predictions," in *Proc. IEEE ICC'10*, May 2010.
 - [76] M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Trans. Veh. Technol.*, vol. 29, pp. 317–325, 1980.
 - [77] H. Holma and A. Toskala, Eds., *WCDMA for UMTS: Radio Access for Third Generation Mobile Communications*, 1st ed. John Wiley & Sons, Inc., 2001.
 - [78] L. Huang and M. Neely, "Utility optimal scheduling in energy harvesting networks," in *Proc. ACM MobiHoc'11*, May 2011.
 - [79] S. Hussmann and D. Jensen, "Crazy car race contest: Multicourse design curricula in embedded system design," *IEEE Trans. Educ.*, vol. 50, no. 1, pp. 61–67, 2007.
 - [80] M. Ibrahim and M. Youssef, "CellSense: An accurate energy-efficient GSM positioning system," *IEEE Trans Vehic. Techn.*, vol. 61, no. 1, pp. 286–296, Jan. 2012.
 - [81] IEEE, "IEEE std. 802.15.4a-2007," 2007.
 - [82] IMEC, "Ultralow power wireless communication," 2013, http://www2.imec.be/be_en/research/wireless-communication/ultralow-power-wireless-communic.html.

-
- [83] Infinite Power Solutions, “Thinergy Micro-Energy Cells,” 2014, <http://www.infinitepowersolutions.com/products/>.
 - [84] K. Jagannathan, I. Menache, E. Modiano, and G. Zussman, “Non-cooperative spectrum access - the dedicated vs. free spectrum choice,” *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 11, pp. 2251–2261, Dec. 2012.
 - [85] J. Jia, Q. Zhang, and X. Shen, “HC-MAC: A hardware-constrained cognitive MAC for efficient spectrum management,” *IEEE J. Sel. Areas Commun.*, vol. 26, no. 1, pp. 106–117, Jan. 2008.
 - [86] X. Jiang, J. Polastre, and D. Culler, “Perpetual environmentally powered sensor networks,” in *Proc. IEEE IPSN’05*, Apr. 2005.
 - [87] A. Johansen, T. Sorensen, and P. Bonnet, “Service and experiment: Towards a perpetual sensor network testbed without backchannel,” in *Proc. IEEE MASS’11*, Oct. 2011.
 - [88] A. Kandhalu, K. Lakshmanan, and R. R. Rajkumar, “U-connect: A low-latency energy-efficient asynchronous neighbor discovery protocol,” in *Proc. ACM/IEEE IPSN’10*, Apr. 2010.
 - [89] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, “Power management in energy harvesting sensor networks,” *ACM Trans. Embedded Comput. Syst.*, vol. 6, no. 4, 2007.
 - [90] E. Karapistoli, D. Stratogiannis, G. Tsiropoulos, and F. Pavlidou, “MAC protocols for ultra-wideband ad hoc and sensor networking: A survey,” in *Proc. Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, 2012.
 - [91] F. Kelly, A. Maulloo, and D. Tan, “Rate control in communication networks: shadow prices, proportional fairness and stability,” *J. Oper. Res. Soc.*, vol. 49, pp. 237–252, 1998.
 - [92] C. Kipnis and P. Robert, “A dynamic storage process,” *Stochastic processes and their applications*, vol. 34, no. 1, pp. 155–169, 1990.
 - [93] “KMote-B,” http://www.tinyosmall.com/product_p/100-101.htm, 2014.

-
- [94] D. E. Knuth, *The Art of Computer Programming, Vol. 1 - Fundamental Algorithms*, 3rd ed. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., 1997.
- [95] J. Krumm and E. Horvitz, "Predestination: Where do you want to go today?" *IEEE Computer*, vol. 40, no. 4, pp. 105–107, 2007.
- [96] H. Kushner and P. Whiting, "Convergence of proportional-fair sharing algorithms under general conditions," *IEEE Trans. Wireless Commun.*, vol. 3, no. 4, pp. 1250–1259, July 2004.
- [97] J. Le Boudec, R. Merz, B. Radunovic, and J. Widmer, "DCC-MAC: a decentralized MAC protocol for 802.15.4a-like UWB mobile ad-hoc networks based on dynamic channel coding," in *Proc. BROADNETS*, Oct. 2004.
- [98] C. S. Lee, J. H. Su, K. E. Lin, J. H. Chang, and G. H. Lin, "A project-based laboratory for learning embedded system design with industry support," *IEEE Trans. Educ.*, vol. 53, no. 2, pp. 173–181, 2010.
- [99] E. A. Lee, "Cyber physical systems: Design challenges," in *International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing (ISORC)*, May 2008, invited Paper.
- [100] J.-S. Lee, Y.-W. Su, and C.-C. Shen, "A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi," in *Proc. IEEE IECON'07*, Nov. 2007.
- [101] Z. Li, W. Chen, C. Li, M. Li, X. Li, and Y. Liu, "FLIGHT: Clock calibration using fluorescent lighting," in *Proc. ACM MobiCom'12*, Aug. 2012.
- [102] R.-S. Liu, P. Sinha, and C. E. Koksal, "Joint energy management and resource allocation in rechargeable sensor networks," in *Proc. IEEE INFOCOM'10*, Mar. 2010.
- [103] V. Liu, A. Parks, V. Talla, S. Gollakota, D. Wetherall, and J. Smith, "Ambient backscatter: wireless communication out of thin air," in *Proc. ACM SIGCOMM'13*, Aug. 2013.

-
- [104] X. Liu, A. Sridharan, S. Machiraju, M. Seshadri, and H. Zang, “Experiences in a 3G network: interplay between the wireless channel and applications,” in *Proc. ACM MobiCom’08*, Sept. 2008.
 - [105] X. Long and B. Sikdar, “A wavelet based long range signal strength prediction in wireless networks,” in *Proc. IEEE ICC’08*, May 2008.
 - [106] “Lorien,” <http://lorien.xrna.net>, 2014.
 - [107] Z. Low, J. Cheong, and C. Law, “Low-cost PCB antenna for UWB applications,” *Antennas Wireless Propag. Lett.*, vol. 4, pp. 237–239, Dec. 2005.
 - [108] R. Margolies, M. Gorlatova, J. Sarik, P. Kinget, I. Kymissis, and G. Zussman, “Project-based learning within a large-scale interdisciplinary research effort,” *CoRR*, vol. abs/1410.6935, 2014. [Online]. Available: <http://arxiv.org/abs/1410.6935>
 - [109] R. Margolies, M. Gorlatova, J. Sarik, G. Stanje, J. Zhu, B. Vigrham, M. Szczodrak, L. Carloni, P. Kinget, I. Kymissis, and G. Zussman, “Energy harvesting active networked tags (EnHANTs): Prototyping and experimentation,” *Submitted to ACM Transactions on Sensor Networks (TOSN)*, 2014.
 - [110] R. Margolies, G. Grebla, T. Chen, D. Rubenstein, and G. Zussman, “Blink: Neighbor discovery on a power harvesting budget,” *submitted*, July 2015.
 - [111] —, “System and method for asynchronous discovery between ultra-low power wireless networked devices,” *Provisional Patent*, 2015.
 - [112] R. Margolies, L. Pena, K. Kim, Y. Kim, M. Wang, M. Gorlatova, J. Sarik, J. Zhu, P. Kinget, I. Kymissis, and G. Zussman, “Demo: An adaptive testbed of energy harvesting active networked tags (EnHANTs) prototypes,” in *Proc. IEEE INFOCOM’13*, Apr. 2013.
 - [113] R. Margolies, A. Sridharan, V. Aggarwal, R. Jana, N. Shankaranarayanan, V. Vaishampayan, and G. Zussman, “Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms,” in *Proc. IEEE INFOCOM’14*, to appear, Apr. 2014.

-
- [114] —, “Exploiting mobility in proportional fair cellular scheduling: Measurements and algorithms,” *IEEE/ACM Trans. Netw.*, 2015, to appear.
- [115] M. J. McGlynn and S. A. Borbash, “Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks,” in *Proc. ACM MobiHoc’01*, Oct. 2001.
- [116] R. Metz, “The Internet of you,” *MIT Technology Review*, May 2014. [Online]. Available: <http://www.technologyreview.com/news/527386/the-internet-of-you/>
- [117] N. Michelusi, K. Stamatiou, and M. Zorzi, “On optimal transmission policies for energy harvesting devices,” in *Proc. Information Theory and Application (ITA) workshop*, Feb.
- [118] J. E. Mills and D. F. Treagust, “Engineering education: Is problem-based or project-based learning the answer?” *Australasian Journal of Engineering Education*, vol. 3, pp. 2–16, 2003.
- [119] B. Miscopein and J.-M. Gorce, “Energy efficient TH-UWB preamble detection,” in *Proc. IEEE ICUWB’09*, Sept. 2009.
- [120] National Intelligence Council, “Disruptive civil technologies – six technologies with potential impacts on US interests out to 2025,” Conference Report CR 2008-07, Apr. 2008.
- [121] D. Noh, L. Wang, Y. Yang, H. Le, and T. Abdelzaher, “Minimum variance energy allocation for a solar-powered sensor system,” in *Proc. DCOSS’09*, 2009.
- [122] P. Papadimitratos, A. La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, “Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation,” *IEEE Commun. Mag.*, vol. 47, no. 11, pp. 84–95, Nov. 2009.
- [123] J. Paradiso and T. Starner, “Energy scavenging for mobile and wireless electronics,” *IEEE Pervasive Comput.*, vol. 4, no. 1, pp. 18–27, 2005.

-
- [124] J. Peckham, P. Stephenson, J. Hervé, R. Hutt, and M. Encarnação, “Increasing student retention in computer science through research programs for undergraduates,” *ACM SIGCSE Bull.*, vol. 39, no. 1, pp. 124–128, 2007.
- [125] P. Peumans, A. Yakimov, and S. Forrest, “Small molecular weight organic thin-film photodetectors and solar cells,” *J. Appl. Phys.*, vol. 93, 2003.
- [126] J. A. Polack-Wahl and K. Anewalt, “Learning strategies and undergraduate research,” *ACM SIGCSE Bull.*, vol. 38, no. 1, pp. 209–213, 2006.
- [127] PowerFilm Solar, “SP3-37 Solar Cell,” <http://www.powerfilmsolar.com/products/oem/sp3-37>, 2014.
- [128] “Preparing the plexcore pv 1000 ink system,” <http://www.sigmaaldrich.com/technical-documents/protocols/materials-science/preparing-the-plexcore.html>, 2014.
- [129] A. Purohit, B. Priyantha, and J. Liu, “WiFlock: Collaborative group discovery and maintenance in mobile sensor networks,” in *Proc. ACM/IEEE IPSN’11*, Apr. 2011.
- [130] L. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. Prentice Hall, 1993.
- [131] M. Radi, B. Dezfouli, K. A. Bakar, S. A. Razak, and M. Lee, “Network initialization in low-power wireless networks: a comprehensive study,” *The Computer Journal*, vol. 57, no. 8, pp. 1238–1261, 2014.
- [132] V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava, “Design considerations for solar energy harvesting wireless embedded systems,” in *Proc. IEEE IPSN’05*, Apr. 2005.
- [133] D. S. Raicu and J. D. Furst, “Enhancing undergraduate education: A REU model for interdisciplinary research,” *ACM SIGCSE Bull.*, vol. 41, no. 1, pp. 468–472, 2009.
- [134] T. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, 2001.

-
- [135] E. P. Saliklis, ““putting a fence” around architectural engineering undergraduate research projects,” in *ASEE National Conference*, June 2006.
- [136] Samsung Electronics Co., “Samsung Galaxy S II,” <http://www.samsung.com/global/microsite/galaxys2/html/specification.html>.
- [137] *Amorphous Solar Cell AM-1815*, SANYO, July 2008, <http://www.farnell.com/datasheets/87124.pdf>.
- [138] J. Sarik, K. Kim, M. Gorlatova, I. Kymissis, and G. Zussman, “More than meets the eye - a portable measurement unit for characterizing light energy availability,” in *Proc. IEEE GlobalSIP’13 Symp. on Energy Harvesting and Green Wireless Communications*, Dec. 2013.
- [139] R. N. Savage, K. C. Chen, and L. Vanasupa, “Integrating project-based learning throughout the undergraduate engineering curriculum,” *Journal of STEM Education: Innovations & Research*, vol. 8, no. 3/4, p. 15, July-Dec. 2007.
- [140] S. Schocken, “Taming complexity in large-scale system projects,” in *Proc. ACM SIGCSE’12*, Feb. 2012.
- [141] A. Schulman, V. Navda, R. Ramjee, N. Spring, P. Deshpande, C. Grunewald, K. Jain, and V. N. Padmanabhan, “Bartendr: a practical approach to energy-aware cellular data scheduling,” in *Proc. ACM MobiCom’10*, Sept. 2010.
- [142] A. Shukla, B. Williamson, J. Burns, E. Burbidge, A. Taylor, and D. Robinson, “A study for the provision of aggregation of frequency to provide wider bandwidth services,” QinetiQ, Tech. Rep., Aug. 2006.
- [143] P. Smith, “Comparing low-power wireless technologies,” <http://www.digikey.com/en/articles/techzone/2011/aug/comparing-low-power-wireless-technologies>, 2011.

-
- [144] S. Solda, M. Caruso, A. Bevilacqua, A. Gerosa, D. Vogrig, and A. Neviani, "A 5 Mb/s UWB-IR transceiver front-end for wireless sensor networks in 0.13 CMOS," *IEEE J. Solid-State Circuits*, vol. 46, no. 7, pp. 1636–1647, July 2011.
- [145] G. Stanje, P. Miller, J. Zhu, A. Smith, O. Winn, R. Margolies, M. Gorlatova, J. Sarik, M. Szczodrak, B. Vigraham, L. Carloni, P. Kinget, I. Kymissis, and G. Zussman, "Demo: Organic solar cell-equipped energy harvesting active networked tag (EnHANT) prototypes," in *Proc. ACM SenSys'11*, Nov. 2011, Best Student Demo Award.
- [146] A. Stolyar, "On the asymptotic optimality of the gradient scheduling algorithm for multiuser throughput allocation," *Operations Research*, vol. 53, pp. 12–25, 2005.
- [147] T. Su, H. Ling, and W. Vogel, "Markov modeling of slow fading in wireless mobile channels at 1.9 GHz," *IEEE Trans. on Antennas and Propag.*, vol. 46, no. 6, pp. 947–948, June 1998.
- [148] W. Sun, Z. Yang, X. Zhang, and Y. Liu, "Energy-efficient neighbor discovery in mobile ad hoc and wireless sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 16, no. 3, pp. 1448–1459, Mar. 2014.
- [149] W. Sun, Z. Yang, K. Wang, and Y. Liu, "Hello: A generic flexible protocol for neighbor discovery," in *Proc. IEEE INFOCOM'14*, Apr. 2014.
- [150] M. Szczodrak and L. Carloni, "Demo: A complete framework for programming event-driven, self-reconfigurable low power wireless networks," in *Proc. ACM SenSys'11*, Nov. 2011.
- [151] J. Taneja, J. Jeong, and D. Culler, "Design, modeling, and capacity planning for micro-solar power sensor networks," in *Proc. IEEE IPSN'08*, Apr. 2008.
- [152] The Physics Arxiv Blog, "Human motion will power the Internet of things, say energy harvesting engineers," *MIT Technology Review*, Jul. 2013. [Online]. Available: <http://m.technologyreview.com/view/516816/human-motion-will-power-the-internet-of-things-say-energy-harvesting-engineers/>

-
- [153] P. Thoppay, C. Dehollain, and M. Declercq, “A 7.5mA 500 MHz UWB receiver based on super-regenerative principle,” in *Proc. ESSCIRC’08*, Sept. 2008.
- [154] “Texas Instruments MSP430 Microcontroller,” www.ti.com/msp430, 2014.
- [155] Time Domain, “PulsON 400 RCM,” <http://www.timedomain.com/p400.php>, 2013.
- [156] “TinyOS,” <http://www.tinyos.net/>, 2014.
- [157] K. Tutuncuoglu, B. Varan, and A. Yener, “Optimum transmission policies for energy harvesting two-way relay channels,” in *Proc. IEEE ICC’13*, June 2013.
- [158] S. Ulukus, A. Yener, E. Erkip, O. Simeone, M. Zorzi, P. Grover, and K. Huang, “Energy harvesting wireless communications: A review of recent advances,” *IEEE J. Sel. Areas Commun.*, vol. 33, no. 3, pp. 360–381, March 2015.
- [159] C. M. Vigorito, D. Ganesan, and A. G. Barto, “Adaptive control of duty cycling in energy-harvesting wireless sensor networks,” in *Proc. IEEE SECON’07*, June 2007.
- [160] B. Vignham and P. Kinget, “A self-duty-cycled and synchronized UWB pulse-radio receiver SoC with automatic threshold-recovery based demodulation,” *IEEE J. Solid-State Circuits*, vol. 49, no. 3, pp. 581–594, March 2014.
- [161] J. Wang and D. Katabi, “Dude, where’s my card?: RFID positioning that works with multipath and non-line of sight,” in *Proc. ACM SIGCOMM’13*, Aug. 2013.
- [162] M. Wang, L. Pena, J. Sarik, H. Wang, R. Margolies, M. Gorlatova, G. Burrow, B. Vignham, P. Kinget, I. Kymissis, and G. Zussman, “Energy harvesting active networked tags (EnHANTs) prototypes,” Nov. 2012, Demo presented in IDtechEx Energy Harvesting and Storage, Washington, DC.
- [163] Z. Wang, A. Tajer, and X. Wang, “Communication of energy harvesting tags,” *IEEE Trans. Commun.*, vol. 60, no. 4, pp. 1159–1166, Apr. 2012.
- [164] E. Wenderholm, “Challenges and the elements of success in undergraduate research,” *ACM SIGCSE Bull.*, vol. 36, no. 4, pp. 73–75, 2004.

-
- [165] D. Wentzloff, F. Lee, D. Daly, M. Bhardwaj, P. Mercier, and A. Chandrakasan, “Energy efficient pulsed-UWB CMOS circuits and systems,” in *Proc. IEEE ICUWB’07*, Sept. 2007.
 - [166] M. Win and R. Scholtz, “Ultra-wide bandwidth time-hopping spread-spectrum impulse radio for wireless multiple-access communications,” *IEEE Trans. Commun.*, vol. 48, no. 4, pp. 679–689, Apr. 2000.
 - [167] W. Wolf and J. Madsen, “Embedded systems education for the future,” *Proc. of the IEEE*, vol. 88, no. 1, pp. 23–30, 2000.
 - [168] Y. Wu, Z. Mao, S. Fahmy, and N. Shroff, “Constructing maximum-lifetime data gathering forests in sensor networks,” *IEEE/ACM Trans. Netw.*, vol. 18, no. 5, pp. 1571–1584, 2010.
 - [169] Y. Yang, L. Wang, D. K. Noh, H. K. Le, and T. F. Abdelzaher, “SolarStore: Enhancing data reliability in solar-powered storage-centric sensor networks,” in *Proc. ACM MobiSys’09*, June 2009.
 - [170] J. Yao, S. S. Kanhere, and M. Hassan, “An empirical study of bandwidth predictability in mobile computing,” in *Proc. ACM WiNTECH’08*, Sept. 2008.
 - [171] —, “Geo-intelligent traffic scheduling for multi-homed on-board networks,” in *Proc. ACM MobiArch’09*, June 2009.
 - [172] L. Yerva, A. Bansal, B. Campbell, T. Schmid, and P. Dutta, “Grafting energy-harvesting leaves onto the sensornet tree,” in *Proc. IEEE IPSN’12*, Apr. 2012.
 - [173] L. You, Z. Yuan, P. Yang, and G. Chen, “ALOHA-like neighbor discovery in low-duty-cycle wireless sensor networks,” in *Proc. IEEE WCNC’11*, Mar. 2011.
 - [174] Y. Yuan, P. Bahl, R. Chandra, T. Moscibroda, and Y. Wu, “Allocating dynamic time-spectrum blocks in cognitive radio networks,” in *Proc. ACM MobiHoc’07*, Sept. 2007.
 - [175] Zebra Technologies, “Dart ultra-wide band UWB,” <http://www.zebra.com/us/en/solutions/technology-need/uwb-solutions.html>, 2013.

-
- [176] J. Zhang, P. Orlik, Z. Sahinoglu, A. Molisch, and P. Kinney, “UWB systems for wireless sensor networks,” *Proc. IEEE*, vol. 97, no. 2, pp. 313–331, Feb. 2009.
 - [177] P. Zhang and D. Ganesan, “Enabling bit-by-bit backscatter communication in severe energy harvesting environments,” in *Proc. USENIX NSDI’14*, Apr. 2014.
 - [178] Y. Zhang, R. Yu, M. Nekovee, Y. Liu, S. Xie, and S. Gjessing, “Cognitive machine-to-machine communications: visions and potentials for the smart grid,” *IEEE Netw.*, vol. 26, no. 3, pp. 6–13, May 2012.
 - [179] J. Zhu, G. Stanje, R. Margolies, M. Gorlatova, J. Sarik, Z. Noorbhaiwala, P. Miller, M. Szczodrak, B. Vignaham, L. Carloni, P. Kinget, I. Kymissis, and G. Zussman, “Demo: Prototyping UWB-enabled EnHANTs,” in *Proc. ACM MobiSys’11*, June 2011.
 - [180] T. Zhu, Z. Zhong, Y. Gu, T. He, and Z.-L. Zhang, “Leakage-aware energy synchronization for wireless sensor networks,” in *Proc. ACM MobiSys’09*, June 2009.
 - [181] A. L. Zydney, J. S. Bennett, A. Shahid, and K. W. Bauer, “Impact of undergraduate research experience in engineering,” *Journal of Engineering Education*, vol. 91, no. 2, pp. 151–157, 2002.

Appendix

Project-based Learning within a Large-Scale Interdisciplinary Research Effort

The modern engineering landscape increasingly requires a range of skills to successfully integrate complex systems. Project-based learning is used to help students build professional skills. However, it is typically applied to small teams and small efforts. This paper describes an experience in engaging a large number of students in research projects within a multi-year interdisciplinary research effort. The projects expose the students to various disciplines in Computer Science (embedded systems, algorithm design, networking), Electrical Engineering (circuit design, wireless communications, hardware prototyping), and Applied Physics (thin-film battery design, solar cell fabrication). While a student project is usually focused on one discipline area, it requires interaction with at least two other areas. Over 5 years, 180 semester-long projects have been completed. The students were a diverse group of high school, undergraduate, and M.S. Computer Science, Computer Engineering, and Electrical Engineering students. Some of the approaches that were taken to facilitate student learning are real-world system development constraints, regular cross-group meetings, and extensive involvement of Ph.D. students in student mentorship and knowledge transfer. To assess the approaches, a survey was conducted

among the participating students. The results demonstrate the effectiveness of the approaches. For example, 70% of the students surveyed indicated that working on their research project improved their ability to function on multidisciplinary teams more than coursework, internships, or any other activity.

7.1 Introduction

The modern engineering landscape requires system engineering skills and interdisciplinary knowledge that are best acquired through participation in large-scale projects. While the need to engage students in large-scale “system perspective” projects has been recognized [140,167], such projects are rarely attempted in academic settings. Project-based learning [79,98] is actively used to help students build professional skills, such as teamwork and communication skills. Despite the many project-based learning approaches that have been attempted and the many frameworks that have been proposed [28,41,133,164], to the best of our knowledge, project-based learning is typically only applied to small teams and small efforts.

In this paper we describe our experience in engaging a large and diverse group of students in project-based learning within a large-scale interdisciplinary research effort. Our experience with organizing multiple student projects to contribute to a large-scale effort is unique and this paper describes our approaches and some of the lessons we learned.

Our ongoing project-based learning activities are related to the “*Internet of Things*” [60] – digital networking of everyday objects. Since 2009 a team of seven faculty members from the Department of Computer Science and the Department of Electrical Engineering at Columbia University have participated in the *Energy Harvesting Active Networked Tags (EnHANTs)* project [6]. The goal of the project is to develop a new type of a networked wireless device. These small, flexible tags (the envisioned form factor for a future EnHANT is shown in Fig. 1.2(a), and a recent mock-up is shown in Fig. 1.2(b)) will be attached to commonplace objects to allow them to communicate and network. EnHANTs will enable futuristic applications envisioned for the Internet of Things, such as finding lost objects (i.e., lost keys, sunglasses, or toys will be accessible via wireless links) and detecting object configurations.

Recent advances in ultra-low-power wireless communications, energy harvesting (deriving energy from ambient sources such as light and motion), and energy harvesting-adaptive networking will enable the realization of EnHANTs in the near future [60]. Designing adaptive networks of EnHANTs requires reconsideration of protocols on all levels of the networking stack. Additionally, designing the EnHANTs to achieve the desired form factor requires tight integration of the networking and communications protocols with the enabling hardware technologies. This necessitates close and continuous interactions of students and faculty members with expertise in the associated technology areas. Working on the EnHANTs project exposes the students to various disciplines in Electrical Engineering (circuit design, wireless communications), Computer Science (embedded systems, algorithm design, networking), and Applied Physics (battery and solar cell design).

Over 17 semesters, we have involved a diverse population of 80 high school, undergraduate, Masters, and Ph.D. students in 180 semester-long research projects related to the design and development of the EnHANT prototypes and the prototype testbed. The student projects are multidisciplinary. A project typically focuses on one disciplinary area, but requires interaction with at least two other areas. The projects necessitate collaboration, provide students with in-depth fundamental understanding of networking concepts, and require students to improve their communication skills. We use “real-world” system integration deadlines and frequent system demonstrations to motivate students and to encourage cross-disciplinary collaboration. Students demonstrated prototypes and the testbed at six conference demonstration sessions [62, 66, 112, 145, 162, 179] and at over three dozen additional live on-site and off-site demonstrations. Students also contributed to publications describing the prototypes and the testbed [61, 109]. To evaluate our learning activities, we conducted a survey among the students. Of the students who completed the survey, over 90% indicated that the project was rewarding and enriching, and 70% indicated that working on this project improved their ability to function on multidisciplinary teams *more than any other activity in their academic career*.

This paper is organized as follows. Section 7.2 describes the educational objectives and the related work. Section 7.3 describes the EnHANTs umbrella project and the student projects. Section 7.4 describes our approaches to organizing student projects and some lessons we learned

over the 5-year course of the umbrella project. Section 7.5 presents the evaluation results and Section 7.6 describes the outcomes of the project. Section 7.7 concludes the chapter.

7.2 Context and Intended Outcomes

Previous research has described structuring student research experiences as a course [72, 126], a research-based program aimed at undergraduates [124], and a framework for accommodating undergraduate students in a research group [28, 164]. Researchers have also examined methods for providing students with interdisciplinary research opportunities [133] and for increasing student communication and collaborative skills [41]. To the best of our knowledge, our experience with providing many students interdisciplinary project-based research opportunities as part of a single large-scale ongoing research effort is unique.

The EnHANTs research effort is in the embedded systems domain. The applicability of project-based learning to embedded systems has been specifically noted. Additionally, the importance of learning embedded system design from a system perspective has previously been emphasized [167]. Project-based learning approaches for embedded system design have also been embraced in different courseworks [79, 98]. The necessity of engaging students in large system development projects has been recognized as an important educational objective, and some tools for emulating the scale of the development have been proposed [140].

This chapter builds on the previous research by describing a method for engaging students in project-based learning within a large-scale multidisciplinary research effort, where the structure of the projects is critical to success [135]. Project-based learning encourages system level thinking [139], helps develop engineering design skills [54] and “soft skills” [74], and is well-suited for team-based [58] and long-term projects [118]. The research projects can help students to develop technical skills and encourage undergraduates to continue to graduate studies [181].

The high-level educational goal of the project is to improve the student’s overall education by exposing them to a large-scale multidisciplinary research project. The project’s main educational objectives focus on improving the students’ abilities to function on multidisciplinary teams and communicate effectively. Specifically, some notable objectives were to enable stu-

dents to effectively understand a *systems* view of their specific project in the context of the overarching umbrella project. It was also our goal to enable students to effectively communicate with team members of different backgrounds (i.e., hardware and software) as part of a unique multidisciplinary design and development process. Other learning outcomes, some of which were inspired by ABET EC 2000 [11] included improving the students ability to design and conduct experiments and analyze and interpret data. In Section 7.5 we discuss evaluation results related to some of these objectives.

7.3 Project-based Learning

The Energy Harvesting Active Networked Tags (EnHANTs) umbrella project [6] has engaged, over 17 semesters, 80 students in 180 interdisciplinary hands-on semester-long projects. In this section, we first briefly describe the umbrella project. Then, we describe the student involvement.

7.3.1 Umbrella Project

The EnHANTs project [6] is a large multi-year interdisciplinary research project related to the *Internet Of Things*. The main focus of the project is the development of the EnHANTs prototypes and the EnHANT prototype testbed. The current prototype is shown in Fig. 1.5. The prototypes harvest indoor light energy using thin-film solar cells, communicate with each other using custom-developed transceivers, and implement custom energy-harvesting-adaptive algorithms on several layers of the Open Systems Interconnection stack. The testbed monitors the prototypes' communications and networking parameters and controls the amount of light incident on the prototypes' solar cells.

The prototypes and the testbed were developed in a set of phases over 5 years. The photos of the testbed in the different phases are shown in Fig. 2.16, which additionally demonstrates the iterative development of the different aspects of the prototype and testbed functionality. At the end of each phase, a fully functional prototype testbed was demonstrated by students, as indicated in Fig. 2.16:

- Phase I demo at IEEE SECON'10 [66]: 7 student coauthors from 2 research groups.

- Phase II demo at ACM MobiCom'10 [62]: 7 student coauthors from 4 research groups.
- Phase III demo at ACM MobiSys'11 [179]: 9 student coauthors from 4 research groups.
- Phase IV demo at ACM SenSys'11 [145]: 10 student coauthors from 4 research groups.
- Phase V demo at IDTechX'12 Energy Harvesting and Storage USA Conference [162]: 10 student coauthors from 3 research groups.
- Phase VI demo at IEEE INFOCOM'13 [112]: 8 student coauthors from 3 research groups.

The Phase IV demo [145] received the conference's *Best Student Demonstration Award*.

The conference demonstrations were conducted in different cities, which necessitated solving not only technological but also logistical challenges. For example, Phase II, IV, and VI demonstrations required shipping. The Phase VI demonstration [112] was conducted abroad, which exposed students to the differences in electrical systems of different countries and to some of the challenges of international shipping logistics.

Additionally, we recently described the prototypes and the testbed in two publications [61, 109] (7 student coauthors from 4 research groups; 16 contributing students from 5 research groups additionally acknowledged).

7.3.2 Student Projects

As of September 2014, over 17 semesters the EnHANTs umbrella project has engaged 80 students who have completed 180 semester-long projects. The number of student projects completed each semester is shown in Fig. 7.1. The student demographics are presented in Fig. 7.2 and Fig. 7.3. Out of 80 students, 6% were high school students we engaged via a Harlem Children Society's program that pairs university researchers with students from under-served communities [7]. 36% were undergraduates, 48% were M.S. students (nearly all of which were in non-thesis terminal M.S. programs), and 10% were Ph.D. students. 73% of students were enrolled in academic programs in Columbia University, while the other 27% were visiting students, such as Research Experience for Undergraduate (REU) students, students from local colleges without advanced research facilities, or visiting international students. Students that were enrolled in academic programs at Columbia University typically took this project as an elective independent research

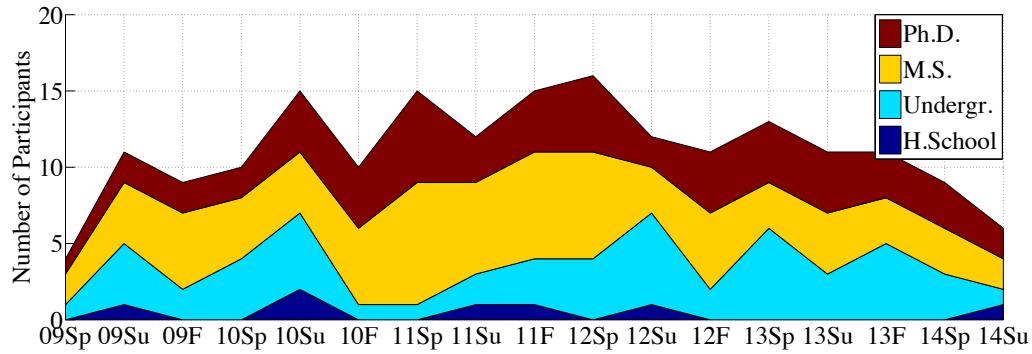


Figure 7.1: Student projects by semester.

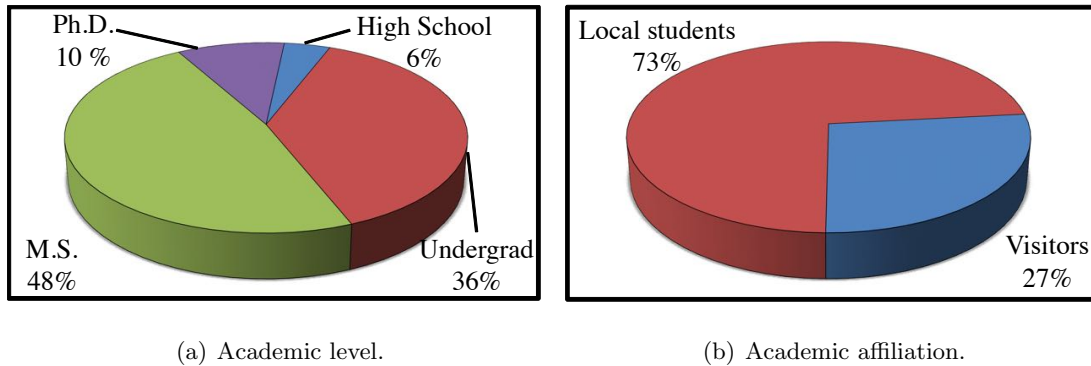


Figure 7.2: Students involved in the EnHANTs project.

project course (separate from the senior design capstone project). Out of the 180 student projects, 51% were full-time projects (summer research internships, REU projects, M.S. thesis research semesters). The other 49% were semester-long research project courses to which students typically dedicated 8-15 hours per week. 72% of the projects were completed by male students and 28% by female students. The main focus areas for the student projects were networking (51%), circuits and systems (17%), and electronics and applied physics (20%). Other projects focused on operating systems and VLSI design (12%).

The student projects within the EnHANTs project are collaborative and multidisciplinary. A project typically focuses on one disciplinary area (e.g., algorithm design, operating systems development, solar cell design) but requires interaction with at least two other areas. These

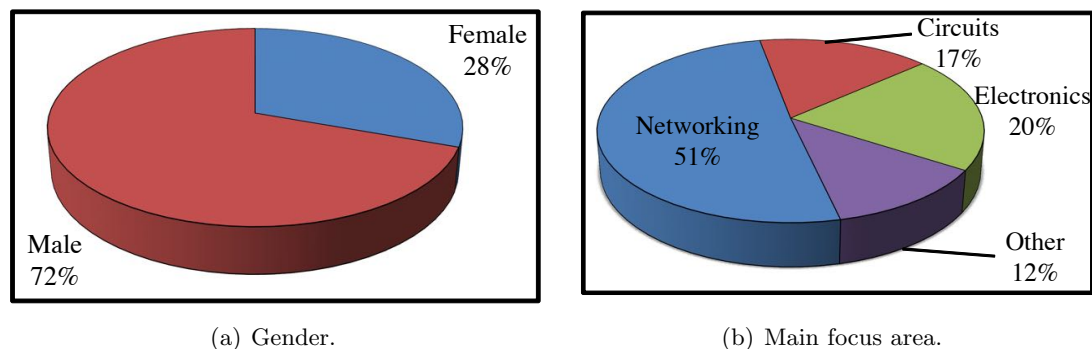


Figure 7.3: Students semesters.

projects challenge students by requiring them to gain understanding of concepts outside of their comfort zone. Additionally, the projects require students to independently and proactively seek out relevant expertise throughout the research groups involved in the EnHANTs project. This improves students' communication and teamwork skills. Four representative student projects are described below. More than 40 additional project descriptions are provided at [6].

- *Real-time monitoring and control system* (Phase I project, completed by an undergraduate Computer Science student): The student developed a Java-based system to monitor and control the EnHANTs prototypes. The project involved designing the necessary data structures to enable communication between the prototypes and the computer running the monitoring system. The student designed the system to support both a text-based interface and a “visual demo” interface that shows the activity of the prototypes in an easy-to-understand way. This project, implemented using TinyOS and Java, required knowledge of sensor networking, wire-line communication, and software design. The student extensively interacted with students who were modifying the prototype operating system and developing energy-harvesting-aware algorithms.
- *Energy harvesting module* (Phase III project, completed by an M.S. Electrical Engineering student): The student improved the design of the Energy Harvesting Module (EHM) that tracked the energy harvested from the solar cell and the state of the battery in real time. The project required the student to improve on a simple prototype that used an off the shelf Coulomb counter. The student worked with the students developing energy-harvesting-

adaptive algorithms to define a new specification for the range, resolution, and response time of the EHM. The student designed and built a new EHM that included a high side current sense amplifier and new digital interface to meet these requirements. This project required knowledge of circuit design and embedded programming and an understanding of how the underlying hardware would affect higher level system performance.

- *Prototype UWB-IR communication module* (Phase III project, completed by an M.S. Electrical Engineering student): The student developed and tested the UWB-IR communication module. The student integrated a custom-designed UWB-IR transmitter and receiver chipset onto a single printed circuit board and programmed a complex programmable logic device to perform data serialization and deserialization, preamble detection, and byte synchronization. The student also developed a UWB-IR radio driver using TinyOS. Primarily focused on circuit design, this project required the student to develop expertise in networking, operating systems, and software design.
- *Energy harvesting-adaptive network* (Phase V project, completed by an undergraduate Computer Engineering student): The student implemented network layer protocols that handled the EnHANT packet routing. The student first tested the network functionality using the commercial transceivers, and then extensively evaluated its performance with the custom UWB-IR transceivers. The student also implemented energy harvesting-adaptive network layer algorithms, which changed packet routing paths based on the environmental energy availability. The student extensively tested these algorithms with the custom energy harvesting modules. While primarily focused on networking, the project required the student to gain an in-depth knowledge of energy harvesting and UWB-IR communications.

7.4 Project Organization and Lessons Learned

Organizing multiple student projects to contribute to a large-scale effort is challenging. In this section we describe how the student projects were structured to facilitate student learning. We also present lessons that were learned throughout the 5-year course of the project. The lessons learned focus on *real-world experiences* including close and continuous cross-group interactions,

well-defined interfaces between student projects, structured short-term goals, real-world work environment, development of student mentors, and project self-promotion.

7.4.1 Multidisciplinary Projects

As described in Section 7.3, the student projects are collaborative and multidisciplinary, often requiring interaction with at least two technical areas. The students work in different labs, focus on different disciplines, and have different technical skills, priorities, work styles, and expectations.

Early on we discovered that the gaps between the knowledge of the students with different expertise areas are much wider than anticipated. For example, Electrical Engineering students are oftentimes unfamiliar with good software development practices, while many Computer Science students may not understand how to handle experimental electronics. Additionally, many students that are not majoring in Electrical Engineering do not understand the concepts of frequency-domain signal processing that are essential to the understanding of wireless networking.

These knowledge gaps often lead to both technological and interpersonal issues. Cross-group problem solving requires students to trust each other's expertise, but these gaps in knowledge can make the trust difficult to establish. When working with students, we encourage close and continuous cross-group interactions to address these issues. We highlight that such gaps are normal and should be treated as learning opportunities.

Additionally, some of the most challenging problems arise when a student is interfacing his or her project with another project. The difficulty of solving these interface problems may lead to interpersonal tensions. In our experience, designing well-defined interfaces between different technologies has been a challenging task that often required faculty members' involvement. Ultimately these issues can only be addressed by establishing, maintaining, and nurturing the connections between the research groups and between the different students.

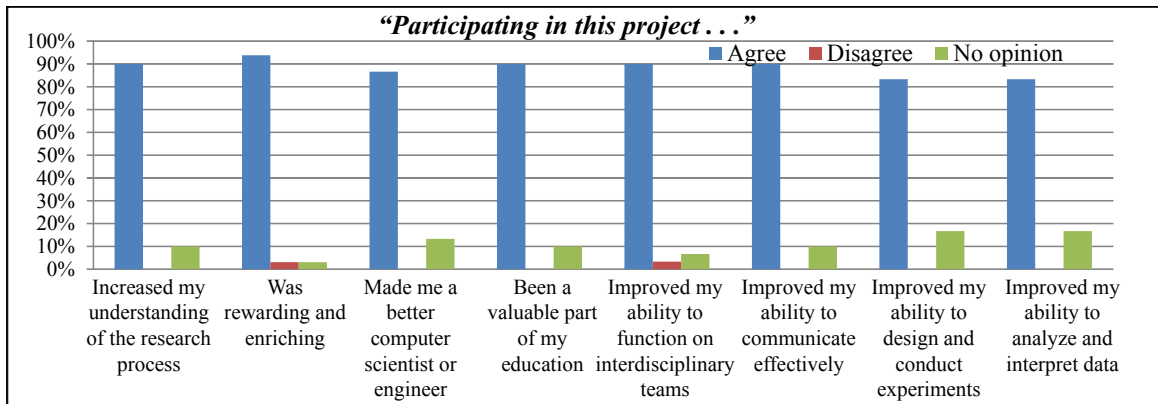


Figure 7.4: Project survey results: Effects of project participation on educational objectives.

7.4.2 Real-world System Integration Deadlines

EnHANTs prototype and testbed design, development, and integration have proceeded in a series of phases (see Fig. 2.16). At the end of each phase, the fully integrated prototype and testbed were presented at a major conference.

Using conference timelines as real-world deadlines for the integration of different student projects has many benefits. Providing structured short-term deadlines for student projects, rather than abstract long-term goals, energizes and motivates the students. Students are additionally motivated by seeing their work integrated with the work of others, used in conference presentations, and subsequently extended. We learned that short-term system integration deadlines also encourage frequent cross-disciplinary collaboration, as the students worked together to quickly solve problems that were arising. In addition, constant updates to the software and the hardware throughout the integration deadlines reduce the impact of unsuccessful projects.

7.4.3 Frequent Cross-group Meetings

Students present their work to the faculty and students from the different research groups at regular (weekly or bi-weekly) meetings. Such frequent cross-group meetings resemble interdepartmental meetings that are used to manage large-scale projects in industry environments. The frequent meetings facilitate cross-group problem-solving and encourage new team members



Figure 7.5: A Phase III conference demonstration [179] presented by M.S. and Ph.D. students.



Figure 7.6: A Ph.D. student mentoring an undergraduate student.

to quickly get up to speed on the state of the projects. Additionally, the presence of faculty and students from different research groups at these meetings challenges students to present their work so that it could be understood by people with different backgrounds, enabling multidisciplinary discussions. The students also report that observing how faculty members solve problems during these meetings improves their own problem solving skills.

7.4.4 Frequent System Demonstrations

Functional “live” EnHANT prototypes and testbed are frequently demonstrated in different on-site and off-site presentations.¹ Frequent demonstrations, particularly those conducted off-site (for example, Fig. 7.5 shows the Phase III conference demonstration setup assembled at the conference location by the student presenters), encourage students to design and develop robust software, hardware, and algorithms and to extensively verify and test their work. This improves students’ technical skills, and provides them with an understanding of the quality standards required from technology in “real-world” applications. Additionally, the testbed demonstrations give students opportunities to present their work to vastly different audiences.

The frequent demonstrations of the EnHANTs system necessitate rigorous risk manage-

¹Inspired by agile software development practices, we ensure that a version of the prototypes and testbed is always ready to be demonstrated. We do not integrate new software or hardware without extensive testing and design for backward compatibility. This further reduces the impact of unsuccessful student projects.

ment. For example, we prepare and fully test 2–3 “hot” backup components for each of the demonstration components. The backups are tested both as individual components and while integrated with the rest of the system. Off-site demonstrations entail rigorous contingency planning. Included in the many logistics to be considered are international shipping of our testbed, prototypes, and monitoring system and voltage and plug adapters to accommodate international power supplies.

7.4.5 Ph.D. Student Mentorship

The faculty members involved in the EnHANTs project are heavily engaged in the student projects. However, faculty members delegate many of the day-to-day student supervision tasks to the Ph.D. students. The Ph.D. students provide technical support and guidance to the students (for example, Fig. 7.6 shows a Ph.D. student troubleshooting a prototype component together with an undergraduate student). Additionally, the Ph.D. students are responsible for testing and verifying junior student projects before integration with EnHANTs prototypes and testbed, and for ensuring continuity between the different student projects.

Handling many of the day-to-day activities, the Ph.D. students quickly learn student mentorship skills from the faculty to become advisors to the high school, undergraduate, and M.S. students. This awards the Ph.D. students with opportunities to practice teaching, assigning deadlines, and motivating their advisees. Additionally, the Ph.D. students serve as initial reviewers for all presentation materials prepared by the junior students, which allows the Ph.D. students to practice giving feedback (both technical and related to different aspects of written and verbal communications). This experience provides the Ph.D. students with insights into the managerial aspects of academic positions, as well as with high level skills that are valuable in industry environments.

7.4.6 Outreach and Promotion

Students also serve as ambassadors for the umbrella project, performing tasks that are oftentimes typically done by faculty. For example, the students often directly present their research

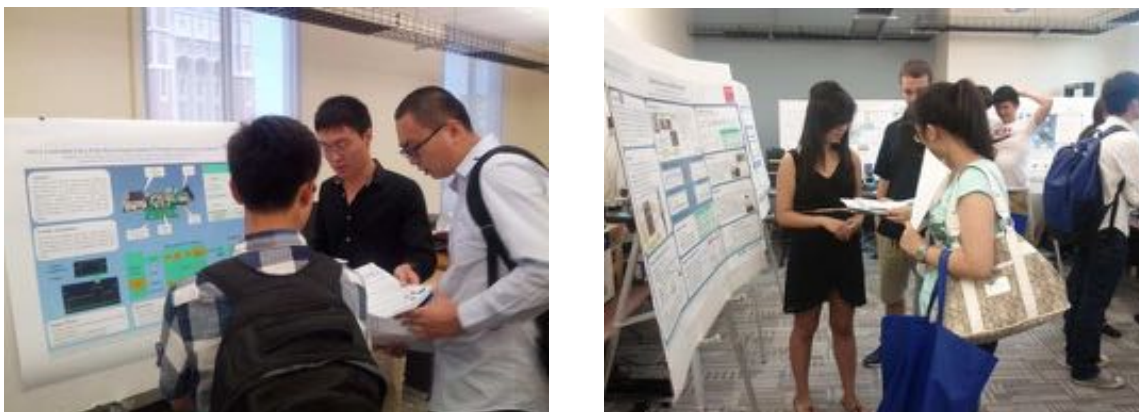


Figure 7.7: Undergraduate and M.S. students presenting their work to incoming M.S. students in Summer 2013.

to prospective graduate and undergraduate students (see Fig. 7.7) and to university visitors. Due to the large scale of the project, the students regularly get opportunities to demonstrate their work to high-profile visitors they would not otherwise encounter. This encourages the students to take pride in showcasing their work. Among the visitors with whom the projects' undergraduate and M.S. students interacted are many top academic researchers from the networking, communications, and electronics communities. Additionally, many visitors came from different companies including IBM, Juniper, Samsung, and Phillips. The students interacted with technical and non-technical company representatives. The non-technical representatives included visitors with roles in marketing, operations, and sales.

Additionally, the students publicize their contributions to the project on the EnHANTs website (see [6] and Fig. 7.8), which is maintained by the members of the project. At the end of each project, the students publish the results of their work on the website. This process encourages students to step back from the intensity of their research, see their work in a positive light, and present it in an exciting manner. We learned that providing students with venues for displaying their work motivated students. The direct involvement of students in the project promotion activities was also useful in recruiting new students.



Figure 7.8: A screenshot of the umbrella project website welcome page [6].

7.4.7 Knowledge Transfer

The large-scale, long-term nature of the umbrella project necessitates knowledge transfer between the students. The project uses an internal wiki that is kept up to date with shared technical information and documentation. We learned that, while knowledge transfer needs to be carefully monitored and emphasized, most students embrace it when they can see first-hand that the documentation they create is used by their peers and mentors. Similarly, most students embrace the opportunity to introduce peers to their work and to teach them.

7.5 Experiences and Feedback

After the completion of Phase V of the umbrella project in October 2012 (see Section 7.3), we conducted a survey to evaluate student learning and the effectiveness of our approaches. In this section, we summarize the results of the survey and other student feedback and briefly discuss our changes to the organizational structure of the umbrella project based on the feedback. We

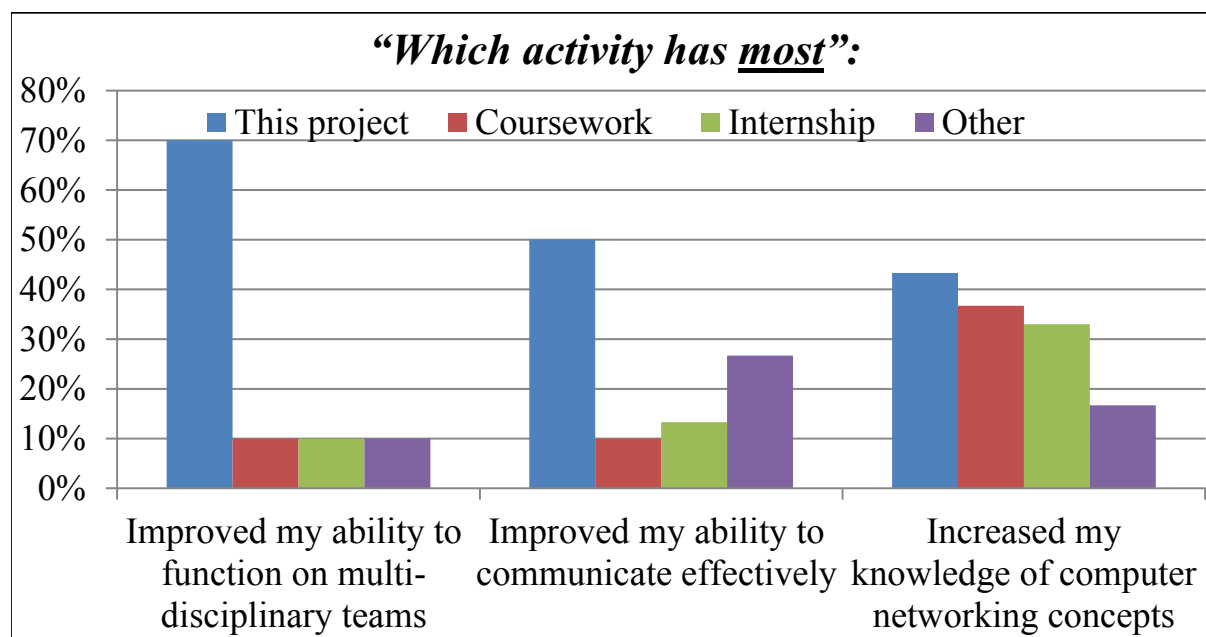


Figure 7.9: Project survey results: Project involvement compared with other activities.

also briefly summarize student outcomes.

We conducted a survey among all 45 high school, undergraduate, and M.S. students that had participated in the project up to October 2012. The survey contained multiple-choice questions and optional open-ended questions. Answers to the multiple-choice questions were measured on a Likert scale: 1 – strongly disagree, 2 – disagree, 3 – neither agree nor disagree, 4 – agree, 5 – strongly agree.

The survey response rate was 75.5%. In the survey’s open-ended questions, students shared many observations, comments, and suggestions about the project organization. Fig. 7.4, 7.9, and 7.10 show some of the results. Fig. 7.4 demonstrates the percentage of students that agreed (or strongly agreed) and disagreed (or strongly disagreed) with statements related to the educational objectives (see Section 7.2) and overall student experience. Fig. 7.9 shows how the students compared their project involvement to other activities (e.g., coursework, internship) on particular aspects of soft skills and technical knowledge. Fig. 7.10 shows on the Likert scale the average evaluation score for some of the survey questions related to the student mentorship by faculty members and Ph.D. students.

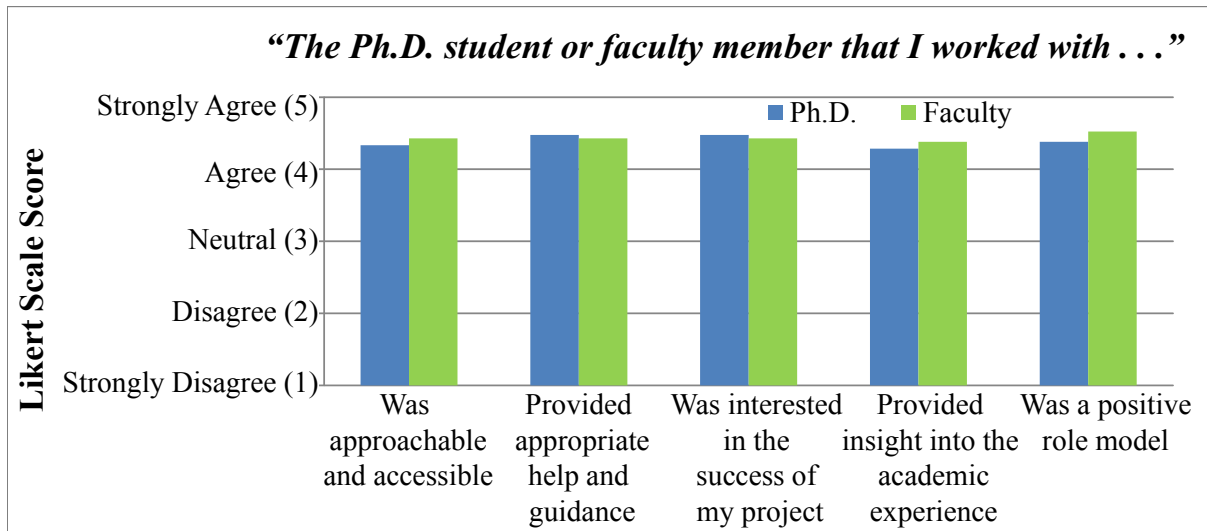


Figure 7.10: Project survey results: Evaluation of mentoring by Ph.D. students and faculty.

Overall, the students' experiences were overwhelmingly positive. Over 90% of the students believed their project experience to be rewarding and enriching. Over 85% of the students indicated that working on this project improved their ability to function on multidisciplinary teams and to communicate effectively, made them a better computer scientist or a better engineer, and was a valuable part of their education (Fig. 7.4). 70% of the students indicated that working on the project improved their ability to function on multidisciplinary teams *more than any other activity*. Additionally, 50% of the students indicated that working on the project improved their ability to communicate effectively more than any other activity, and over 40% of the students indicated that this project increased their knowledge of computer networking more than any other activity (Fig. 7.9).

The students' feedback on the project organization features provided additional insights into the features' effectiveness.

- Multidisciplinary projects:** Most students enjoyed the multidisciplinary nature of their projects. When specifying what they liked most about the project, over 50% of the students commented on one of its multidisciplinary aspects. One student enjoyed her project being *“about both hardware and software”*, and said it was *“innovative and challenging to integrate many different aspects in one”*. One student's favorite thing about his project was the

	Undergraduate	M.S.
Academic	M.S. Students – Columbia (1), University of Michigan (1); Ph.D. Candidates – Columbia (1), Princeton (2), Texas A&M (1), UT Austin (1)	Ph.D. Candidates – Carnegie Mellon (1), Columbia (6), Harvard (1), Northwestern (1)
Software Roles	Program Manager at Microsoft; Manager of Cloud Operations at Bullhorn; Associate Applications Developer at AT&T; Flight Software Engineer at Marshall Space Flight Center Software Developers – Akcelita (1), Bloomberg LP (1), Credit Karma (1), ZocDoc (2)	iOS Developer at Longtail Studios; Cofounder of Software Company (Vessel.io); Technology Associate at JPMorgan Chase; Applications Engineer at OPNET; Sales Engineer at Bluebox Security Software Developers – Apple (1), Interactive Brokers (1), Lam Research (1), Microsoft (1), Oracle (1), VMWare (1), VM Turbo (1), Work Market (1)
Hardware Roles	Battery Test Engineer at 24M Technologies; Product Engineer at Intel; Electrical Engineer at Bala Consulting Engineers	Systems Engineers at Raytheon Co. (2); Electrical Engineer at SunPower Corp. Circuit Design Engineers – Ambarella (1), SK Hynix (1), Credit Suisse (1), Silicon Laboratories (1), Boeing Space Systems (1)
Other	Unknown (3)	Pilot at U.S. Air Force; Business Assistant at TCL Multimedia; Unknown (1)

Table 7.1: The placements of 22 undergraduate and 32 M.S. students that already graduated (6 undergraduate and 5 M.S. students are yet to graduate).

“integration of my work with other parts of the system – felt like a cohesive project that mattered more”. However, students also noted that *“getting everyone on same page was difficult at times”*, and said that *“not being able to know exactly what others are doing”* was an impediment to achieving some of their project goals. Based on this feedback, we have been encouraging the students to independently and proactively collaborate with each other.

- **Real-world system integration deadlines:** While students nearly universally enjoyed being able to integrate their work with the work of others, students’ opinions about the deadlines were mixed. Some of the students said that they liked the *“feel of product evolution”* and *“liked the challenging deadlines”*. Yet other students pointed out several deadline-related inefficiencies in project organization. For example, one of the students noted that at times he was not given enough advance notice on system requirements prior to a deadline. Another

student mentioned that the deadlines resulted in unnecessarily “long and boring” system testing. Another student also felt that because of too many integration deadlines she did not have enough time to make progress on her components’ next-generation design. To improve the student’s experiences, we dedicated extra attention to making sure that our subsequent deadlines (such as the Phase VI deadline in April 2013) were organized such that they did not negatively impact the students.

- **Frequent cross-group meetings:** Most students appreciated the opportunities for problem-solving and work discussions and showcasing provided by the regular cross-group meetings. One student noted that *“the meetings are an excellent way for putting everything in the big picture.”* Yet several students also commented that the meetings were unnecessarily long, and suggested that a better meeting structure should be considered for future projects. To improve the quality of the meeting presentations we encourage students to discuss their presentation with their Ph.D. mentors. We are also considering introducing joint presentations for students from the same research group.
- **Frequent system demonstrations:** *Over 95% of the students* indicated that presenting their work was a rewarding and enriching experience. Several students specifically mentioned the presentation skills among the skills they acquired or improved while working on the umbrella project. One student noted that *“the opportunity to present to others was invaluable. Plus it was a lot of fun!”*
- **Student mentorship:** The majority of students appreciated the support provided by their Ph.D. student mentors. Over 80% of the students said that their mentor was approachable and accessible, and provided appropriate guidance (Fig. 7.10). One student noted: *“I’ve never considered pursuing higher education past undergrad level, but I’m starting to see the appeal in getting a Masters and/or Ph.D. now, after working around so many happy and passionate people.”* Several students additionally specifically complimented their mentors.
- **Outreach and promotion:** Students were enthusiastic and excited about many different aspects of the opportunities to present their work. More than 80% of the students said that this experience improved their communication skills and more than 50% said that it helped them to clarify their career path. One of the students noted that presenting her work

was an *"overall confidence booster"*. Another specifically mentioned that it was *"incredibly rewarding to start the project not knowing very much about the subject and end with the ability to present my work with real knowledge of what I had worked on"*.

- **Knowledge transfer:** Despite students' favorable opinions of their mentors and despite their appreciation of the opportunities to teach others, a large portion of the negative feedback focused on insufficient knowledge transfer. Several students commented on the insufficient technical introduction to their project. One student stated that *"in the beginning I felt I didn't have enough support to ask very basic things"*, and another student noted that *"a lot of work goes to waste if you are unable to successfully pass it on to the next person"*. Based on this feedback we have increased our efforts to ensure that the students create high-quality up-to-date documentation.

Following the feedback from the survey, as we worked through the Phase VI demonstration of the umbrella project and past it, we have been modifying the organizational aspects of the project to additionally facilitate student development. We take added care in scheduling tasks related to system integration deadlines, so as not to over-burden students and detract from research and educational objectives. To address cross-group technological issues more quickly and to improve the effectiveness of weekly cross-group meetings, we also conduct many more frequent (up to 2–3 times per week) smaller-group interactions, typically involving only the relevant students and their Ph.D. mentors. Additionally, we take added care to ease students' introduction to the project and to ensure other aspects of knowledge transfer. For example, we now educate students on important aspects of umbrella project disciplines that they may be unfamiliar with (e.g., we ensure that computer science students are trained to handle experimental electronics).

7.6 Outcomes

In this section, we list the outcomes of the project including project awards, individual awards, student publications, and career placements.

Project Awards: A demonstration of the EnHANTs testbed (with 10 student authors and 4 faculty mentors) received the Best Student Demo Award at the ACM SenSys'11 conference

[145]. The EnHANTs project also won the 1st place in the 2009 Vodafone Americas Foundation Wireless Innovation Competition and a project vision paper (with 3 students acknowledged) received the 2011 IEEE Communications Society Award for Advances in Communications [60]. Following these recognitions, the EnHANTs project and the student outcomes were also featured in several media outlets including BBC/PRI, RFID Journal, Columbia's The Record, IEEE Spectrum, and MIT Technology Review.

Individual Awards: Students received: NSF Graduate Research Fellowship, 2013 MIT EECS Rising Star workshop invitation, Columbia Presidential Teaching Award, Wei Family Foundation Fellowship (stipend and tuition for an incoming Ph.D. student), Exzellenzstipendium 2011 Fellowship (competitive Austrian fellowship), as well as departmental Undergraduate Research Awards (two), and M.S. Awards of Excellence (two). The Ph.D. mentors received several awards, some directly stemming from mentoring activities including the Google Anita Borg Fellowship, NSF Graduate Research Fellowship, Computing Innovation Fellowship as well as departmental Jury Award, Armstrong Award, and Collaborative Research Award.

Publications: Nine students co-authored research papers [12, 30, 61, 63, 67, 70, 138, 163]. A undergraduate authored [63] (with four other students acknowledged). This paper was featured in the MIT Technology Review Report on Business [116] and in the MIT Technology Review Physics ArXiv Blog [152]. 29 students co-authored published demo descriptions [62, 66, 112, 145, 162, 179] (see Section 7.3) and five students co-authored first-of-their-kind energy-harvesting datasets [42, 71].

Career Placement: As of September 2014, 83% of the B.S. and M.S. students have graduated and the other 17% are continuing their studies. Table 7.1 shows the current positions of the graduated students. Of the students who have already graduated, 30% continued to higher-level academic programs in top schools like Princeton, Harvard, and Carnegie Mellon, *including seven at Columbia (some of which later became mentors in the program)*.² The other 70% were recruited by technology companies, such as Microsoft, Raytheon, Boeing, Apple, Intel, OPNET, Oracle, VMturbo, and ZocDoc, as well as by NASA and the US Air Force. Several students

²All of the M.S. students were originally enrolled in an M.S. only program that does not lead to a Ph.D. and the high percentage of Ph.D. acceptance among them is very unique.

have indicated that working on the umbrella project prepared them for some of the challenges they face in their careers. For example, one student noted that the “*experience presenting my work has been really helpful in my current job profile*”, and another highlighted that “*being held accountable for deadlines and project completeness helped prepare me for work environment*”.

In addition, a few of the Ph.D. mentors already graduated and assumed postdoctoral positions in EPFL, Princeton, and Columbia, research positions in AT&T Labs Research and D.E. Shaw Research, and industry positions. Three of the mentors also founded start-ups. All of the mentors indicated that their mentoring experience was instrumental in shaping their career paths.

7.7 Conclusions

While the modern computing landscape increasingly requires large-scale system engineering skills, such skills are rarely acquired in a typical computing program. To address this, over the last 5 years, we have been engaging a diverse group of students in research projects within a large-scale interdisciplinary *Energy Harvesting Active Networked Tags (EnHANTs)* project. To date, 180 semester-long projects have been completed within the EnHANTs project. The projects challenge students’ knowledge and organizational and communication skills. Some of the approaches we have used to facilitate student learning are the real-world system development constraints, regular cross-group meetings, and extensive involvement of Ph.D. students in student mentorship and knowledge transfer. Students find the projects rewarding and gain valuable skills. Our experience demonstrates feasibility of engaging diverse groups of students on large-scale interdisciplinary research efforts. It sheds light on some potential pitfalls of such efforts (e.g., inadequate cross-group communication and knowledge transfer), and suggests best practices to overcome these challenges.